

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Jure Prevc

**Razvoj grafičnega uporabniškega
vmesnika za vizualizacijo, procesiranje
in analizo posnetkov
elektrohisterograma**

DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana, 2016

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Jure Prevc

**Razvoj grafičnega uporabniškega
vmesnika za vizualizacijo, procesiranje
in analizo posnetkov
elektrohisterograma**

DIPLOMSKO DELO
UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: prof. dr. Franc Jager

Ljubljana, 2016

Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Razvijte grafični uporabniški vmesnik za vizualizacijo, procesiranje in analizo posnetkov elektrohisterograma. Vmesnik naj omogoča vizualizacijo signalov v časovnem in frekvenčnem prostoru, izločanje motenj z uporabo Butterworthovih digitalnih filtrov in dvosmerne sheme filtriranja, postavljanje oznak in izrezovanje določenih delov posnetkov, ocenjevanje amplitudnih in močnostnih spektrov signalov z uporabo Fourierjeve transformacije in avto-regresivne metode ter analizo močnostnih spektrov z iskanjem vrha spektra, frekvence vrha spektra in mediane frekvence. Vmesnik naj je razvit v okoljih NetBeans in Java.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Jure Prevc sem avtor diplomskega dela z naslovom:

Razvoj grafičnega uporabniškega vmesnika za vizualizacijo, procesiranje in analizo posnetkov elektrohisterograma

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Franca Jagra,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 4. februarja 2016

Podpis avtorja:

Diplomsko delo, v katerega je bilo vloženo veliko časa in truda, ne bi nastalo brez vse podpore, ki sem jo bil deležen od družine in prijateljev. Iskrena hvala družini, ker ste verjeli vame in me spodbujali pri mojih odločitvah. Posebna zahvala tudi mentorju prof. dr. Francu Jagru, za usmerjanje in strokovno pomoč pri izdelavi diplomskega dela.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Namen dela	1
1.2	Elektrohisterogram in napovedovanje prezgodnjega poroda . .	2
2	Orodja in knjižnice	5
2.1	NetBeans IDE	5
2.2	Programski paket WFDB	7
2.3	Podatkovna baza TPEHG	7
2.4	Gnuplot	9
2.5	Java DSP collection	9
3	Opis metod	11
3.1	Principi načrtovanja uporabniških vmesnikov	11
3.1.1	Nielsenovi principi	11
3.2	Vizualizacija signalov	13
3.3	Filtriranje signalov	14
3.3.1	Butterworthovi filtri	15
3.3.2	Dvosmerna shema filtriranja	17
3.4	Manipulacija posnetkov	18
3.5	Fourierova transformacija	18

KAZALO

3.6	Ozenske funkcije	19
3.7	Amplitudni in močnostni spekter	22
3.8	Avtoregresivna metoda za ocenitev spektra	22
3.9	Analiza močnostnih spektrov	24
3.10	Razvoj grafičnega vmesnika EHGLab	25
3.10.1	Glavno okno	25
3.10.2	Vrstični menu	25
3.10.3	Orodna vrstica	35
3.10.4	Risalna površina	37
3.10.5	Prostor za podrobnosti o signalih	38
3.10.6	Statusna vrstica	38
4	Rezultati	41
4.1	Grafični vmesnik EHGLab in primeri uporabe	41
5	Razprava in zaključki	57

Slike

1.1	Signal EHG zajet v 30. tednu - porod v 39. tednu	4
1.2	Signal EHG zajet v 30. tednu - porod v 32. tednu	4
2.1	Glavno okno Netbeans IDE	6
2.2	Postavitev elektrod na trebuhu	8
2.3	Primer zaglavne datoteke	9
3.1	Teorija upoštevanja vzorca najbližjega piksla	14
3.2	Prenosna karakteristika nizko prepustnih Butterworthovih fil- trov različnih redov	16
3.3	Prenosna karakteristika visoko prepustnih Butterworthovih fil- trov različnih redov	17
3.4	Karakteristike različnih okenskih funkcij	20
3.5	Vzorčni signal pred uporabo okenske funkcije	21
3.6	Vzorčni signal po uporabi okenske funkcije <i>Blackman – Harris</i>	21
3.7	Glavno okno uporabniškega vmesnika EHGLab	26
3.8	Izgled vrstičnega vrstičnega menuja	27
3.9	Dialog za pregledovanje in urejanje zaglavnih datotek	28
3.10	Dialog za izbiro signalov namenjenih za risanje na risalno po- vršino	29
3.11	Dialog za vnos mejnih frekvenc Butterworthovih filtrov	30
3.12	Izvorni signal in trije filtrirani signali	32
3.13	Normaliziran frekvenčni spekter - AR metoda za ocenitev spek- tra	34

3.14	Izgled orodne vrstice	35
3.15	Risalna površina z izrisanimi signali	39
4.1	Papirnati prototip	41
4.2	Dialog za odpiranje posnetka	42
4.3	Prikaz signalov po uspešno prebranem posnetku	43
4.4	Dialog za izbiro izrisanih signalov	44
4.5	Dialog za izbiro mejnih frekvenc digitalnih Butterworthovih filtrov	44
4.6	Izbira filtriranih signalov znotraj uporabniškega vmensika . . .	45
4.7	Prikaz filtriranih signalov znotraj uporabniškega vmesnika . .	46
4.8	Označevanje lokalnega intervala	47
4.9	Prikaz povečanega dela signala znotraj lokalnega intervala . .	47
4.10	Dialog za vnos mejnih frekvenc območja normalizacije	48
4.11	Menu <i>Analysis</i> in izbrana končna izbira za izračun normali- ziranega močnostnega spektra	48
4.12	Normaliziran močnostni spekter filtriranega prvega kanala . .	49
4.13	Dialog za shranjevanje posnetkov v formatu WFDB	50
4.14	Vsi kanali posnetka <i>tpehg1128</i> predstavljeni v časovnem prostoru	51
4.15	Vsi kanali posnetka <i>tpehg1756</i> predstavljeni v časovnem prostoru	51
4.16	Močnostni spekter 1. kanala posnetka <i>tpehg1128</i> - terminski porod	53
4.17	Močnostni spekter 1. kanala posnetka <i>tpehg1756</i> - prezgodnji porod	53
4.18	Močnostni spekter 2. kanala posnetka <i>tpehg1128</i> - terminski porod	54
4.19	Močnostni spekter 2. kanala posnetka <i>tpehg1756</i> - prezgodnji porod	54
4.20	Močnostni spekter 3. kanala posnetka <i>tpehg1128</i> - terminski porod	55
4.21	Močnostni spekter 3. kanala posnetka <i>tpehg1756</i> - prezgodnji porod	55

Seznam uporabljenih kratic

kratica	angleško	slovensko
EHG	Electrohysterogram	elektrohisterogram
IDE	Integrated development environment	Integrirano razvojno okolje
OS	Operating system	Operacijski sistem
JRE	Java Runtime Environment	Java izvajalno okolje
WFDB	WaveForm DataBase	Podatkovni format WFDB
TPEHG DB	Term-Preterm ElectroHysterogram DataBase	Podatkovna baza terminskih in prezgodnjih porodov elektrohisterograma
DSP	Digital signal processing	Digitalno procesiranje signalov
FFT	Fast Fourier Transform	Hitra Fourierjeva transformacija
DFT	Discrete Fourier Transform	Diskretna Fourierjeva transformacija
AR	Autoregression	Avtoregresija

Povzetek

Cilj diplomske naloge je bilo razviti grafični uporabniški vmesnik za vizualizacijo, procesiranje in analizo posnetkov elektrohisterograma (EHG). Razvili smo grafični uporabniški vmesnik EHGLab, prilagojen za delo s posnetki EHG v formatu WaveForm DataBase (WFDB). Vmesnik EHGLab omogoča prikaz signalov v časovnem in frekvenčnem prostoru. Za izločanje motenj v signalih smo načrtali Butterworthove digitalne filtrov z uporabo dvosmerne sheme filtriranja. Uporabniški vmesnik omogoča osnovno manipulacijo nad posnetki EHG, kot je postavljanje oznak in izrezovanje določenih delov posnetkov. Za ocenjevanje amplitudnih in močnostnih spektrov smo uporabili algoritem hitre diskretne Fourierjeve transformacije in avtoregresivno metodo. Vmesnik omogoča tudi analizo izračunanih močnostnih spektrov v izbranem frekvenčnem področju z iskanjem vrha spektra, frekvence vrha spektra in mediane frekvence. Vmesnik je razvit v okoljih NetBeans in Java.

Ključne besede: grafični uporabniški vmesnik, posnetki elektrohisterograma, vizualizacija signalov, ocenjevanje močnostnih spektrov, analiza močnostnih spektrov.

Abstract

The goal of the thesis was to develop a graphic user interface for visualization, processing and analysis of electrohysterogram (EHG) records. We developed the graphic user interface EHGLab, which is adapted for work with the EHG records in the WaveForm DataBase (WFDB) format. The EHGLab interface allows display of the signals in time and frequency domain. To reject noise in signals we designed Butterworth digital filters using bidirectional filtering scheme. The user interface allows basic manipulation of the EHG records, like setting annotations and cutting selected parts of the records. For the estimation of the amplitude and power spectra we used the fast Fourier transformation algorithm and autoregressive method. The interface allows also analysis of the computed power spectra in the selected frequency band by seeking for the peak spectrum, the peak frequency of the spectrum, and the median frequency. The interface was developed in the NetBeans and Java environments.

Keywords: Graphic user interface, Electrohysterogram records, Visualization of signals, Estimation of power spectra, Analysis of power spectra.

Poglavje 1

Uvod

1.1 Namen dela

Namen dela je bilo razviti grafični uporabniški vmesnik za vizualizacijo, procesiranje in analizo posnetkov elektrohisterograma (EHG). Želeli smo razviti specifičen uporabniški vmesnik, ki je zmožen rokovanja s posnetki EHG zapisanih v formatu WaveForm DataBase (WFDB).

Za vizualni prikaz smo posnetke izrisali v časovnem prostoru. Zaradi šumov, ki so lahko prisotni v posnetkih kot posledica okolja ali drugih dejavnikov, smo dodali možnost uporabe filtriranja. Zaradi pozitivnih lastnosti in nekompleksnosti smo uporabili digitalne Butterworthove filtre. Fazni zamik, ki nastane pri uporabi te vrste filtrov smo izničili z uporabo dvosmerne sheme filtriranja. Zadostili smo tudi potrebo po osnovni manipulaciji posnetkov. Znotraj uporabniškega vmesnika je tako mogoče urejanje zaglavnih datotek, postavljanje oznak in izrezovanje določenih delov posnetka. Za vsak signal znotraj posnetka je mogoče izračunati vrste različnih spektrov. Tako imamo znotraj uporabniškega vmesnika možnost izračuna amplitudnega in močnostnega spektra z algoritmom hitre Fourierjeve transformacije (FFT) in avtoregresivne (AR) metode za oceno spektra. Nad vsakim spektrom je mogoče opraviti tudi normalizacijo. Vsak izračunan spekter vizualno prikažemo v novem oknu. Za potrebe analize posnetkov EHG, ki bi lahko pripomogla

k razvoju tehnik pri napovedovanju prezgodnjega poroda smo vsak izračunan spekter ovrednotili z iskanjem vrha spektra, frekvence vrha spektra in mediane frekvence.

Razvoj je potekal nad podatki iz podatkovne baze Term-Preterm ElectroHysteroGram DataBase (TPEHG DB), katera vsebuje posnetke EHG terminskih in prezgodnjih porodov. Podatkovna baza TPEHG DB je na voljo na prosto dostopno spletnem mestu PhysioNet [1]. Ker smo želeli zagotoviti delovanje uporabniškega vmesnika neodvisno od operacijskega sistema, smo za razvoj uporabili programski jezik Java.

1.2 Elektrohisterogram in napovedovanje prezgodnjega poroda

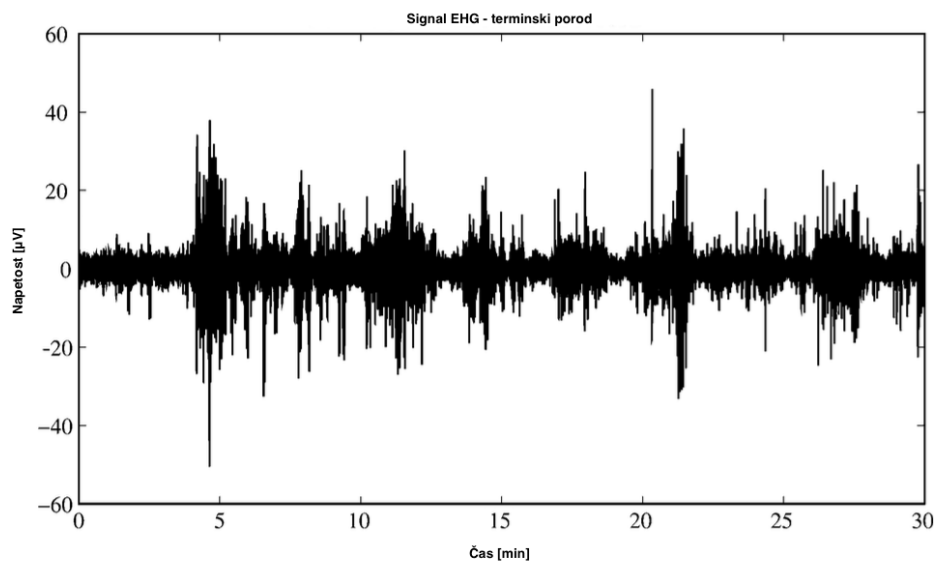
Prezgodaj se na letni ravni rodi okrog 6% otrok. O prezgodnjem porodu govorimo takrat, ko nosečnosti traja manj kot 37 tednov. Relativno majhen odstotek prezgodaj rojenih otrok pa vseeno predstavlja velik problem, saj je prezgodnji porod glavni vzrok obolevnosti in umrljivosti pri nedonošenčkih [2]. Glede na problematiko pa napovedovanje prezgodnjega poroda ni enostavno. Raziskovalci se po vsem svetu trudijo ugotoviti čim bolj učinkovit način napovedovanja prezgodnjega poroda. Eden od načinov je napovedovanje na podlagi dejavnikov tveganja pri materah [3]. Drug način pa bi lahko temeljil na analizi posnetkov EHG.

EHG je zapis mišičnih aktivnosti maternice noseče ženske. Mišično aktivnost zaznavamo s pomočjo električnih odzivov na elektrodah, katere so nameščene na trebuh. Uporaba EHG je mogoča tako med nosečnostjo kot tudi med samim porodom. Na podlagi zajetih signalov EHG lahko zagotovimo zanesljive informacije o popadkih [4]. Študije kažejo, da lahko karakterizacija popadkov pripomore k zgodnji diagnostiki prezgodnjega poroda [5]. Ker je EHG enostavna, popolnoma neboleča tehnika za spremljanje aktivnosti maternice, bi bila uporaba mogoča v vseh porodnišnicah.

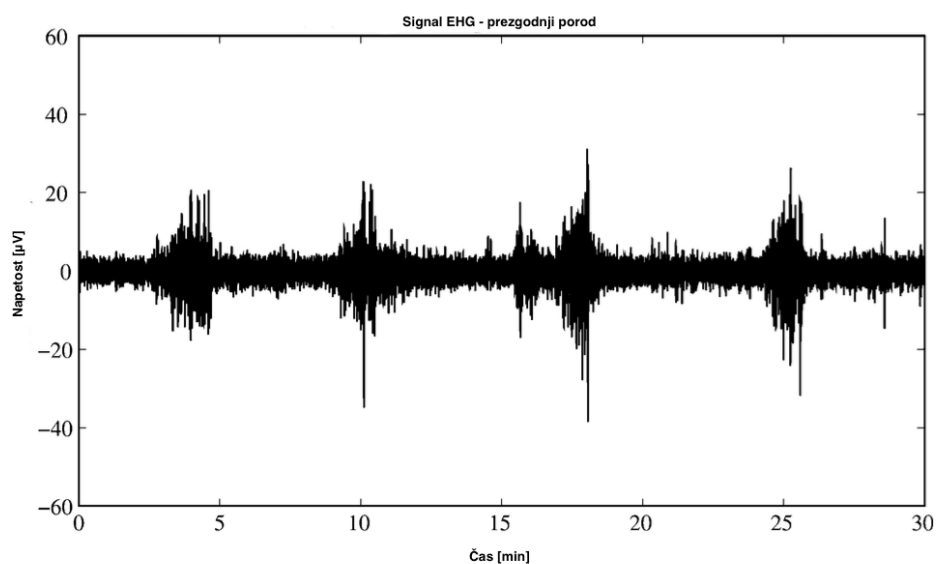
Ena od študij, ki je bila opravljena vzporedno z razvojem podatkovne

baze TPEHG DB [6] nakazuje, da bi lahko na podlagi analize posnetkov EHG razvili tehniko za napovedovanje prezgodnjega poroda. Osredotočili so se na linearne in nelinearne tehnike za analizo posnetkov EHG. Za linearne tehnike so uporabili koren srednje kvadratne vrednosti, prvi prehod preko ničle koeficientov avtokorelacije, frekvenco vrha močnostnega spektra signala in mediano frekvence močnostnega spektra signala. Slednji dve tehniki smo razvili tudi v našem uporabniškem vmesniku. Za nelinearne tehnike analize so se osredotočili na maksimalni eksponent Ljapunova in korelacijsko dimenzijo ter vzorčno entropijo.

Sliki 1.1 in 1.2 iz študije predstavljata po en signal različnih posnetkov EHG. Pri sliki 1.1 je šlo za terminski porod medtem, ko je pri sliki 1.2 šlo za prezgodnji porod. Oba posnetka sta bila zajeta v 30. tednu nosečnosti. Iz slik je na videz vidna razlika o gibanju napetosti skozi čas. Na sliki terminskega poroda 1.1 lahko opazujemo nekoliko bolj kaotičen signal, zdi se, da je v samo mišično aktivnost maternice vpletenih več fizioloških mehanizmov. Opazimo tudi, da je mišična aktivnost težje napovedljiva. Na sliki prezgodnjega poroda 1.2 pa lahko opazimo bolj predvidljivo mišično aktivnost oziroma lažje določimo periodo popadkov. Iz tega razloga je analiza signalov EHG pomembna pri morebitni diagnozi prezgodnjega poroda [6].



Slika 1.1: Signal EHG zajet v 30. tednu - porod v 39. tednu



Slika 1.2: Signal EHG zajet v 30. tednu - porod v 32. tednu

Poglavje 2

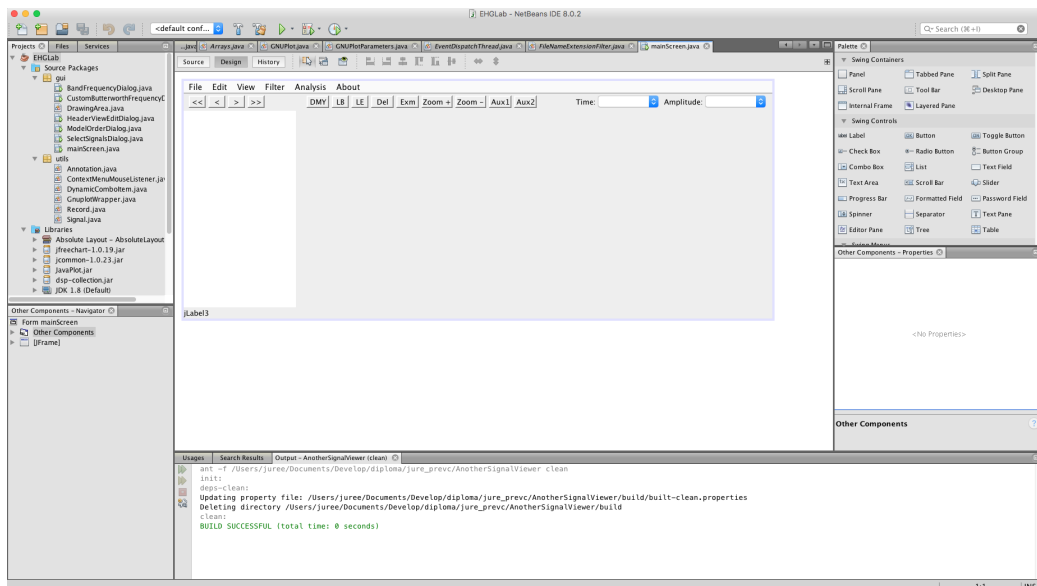
Orodja in knjižnice

2.1 NetBeans IDE

NetBeans IDE je programska oprema, ki razvijalcem nudi vse potrebne komponente za razvoj. NetBeans je odprtokoden in prosto dostopen na spletu [7]. Razvit je v programskem jeziku Java in je na voljo za vse popularne operacijske sisteme kot so npr. Windows, Mac OS X ter Linux. Omogoča razvoj v vrsti programskih jezikov, primarno pa je namenjen razvoju v programskem jeziku Java.

Uporabniški vmesnik je bil razvit s pomočjo NetBeans IDE in programskim jezikom Java. Za grafične gradnike smo uporabili Javi priloženo knjižnico Swing. Java nam tudi omogoča, da razvit uporabniški vmesnik deluje neodvisno od operacijskega sistema. Za zagon programske opreme razvite v Javi potrebujemo Java Runtime Environment (JRE).

Zaslonska slika 2.1 zajeta v operacijskem sistemu Mac OS X prikazuje glavno okno NetBeans IDE.



Slika 2.1: Glavno okno Netbeans IDE

2.2 Programski paket WFDB

Programski paket WFDB [8] je bil razvit za potrebe analize signalov podatkovnih baz shranjenih v arhivu PhysioBank zapisanih v formatu WFDB. [9, 10]. Programski paket je sestavljen iz treh glavnih komponent:

- WFDB knjižnica - množica funkcij za branje in pisanje v format WFDB. Uporaba le teh je možna v v programih izdelanih v programskih jezikih ANSI/ISO C, K&R C, C++ ali Fortran [11].
- WFDB aplikacije - množica ukazno vrstičnih orodij namenjenih za procesiranje in avtomatsko analizo signalov zapisanih v formatu WFDB [12].
- programska oprema WAVE - interaktivni grafični vmesnik za manipuliranje in označevanje posnetkov v formatu WFDB [13].

2.3 Podatkovna baza TPEHG

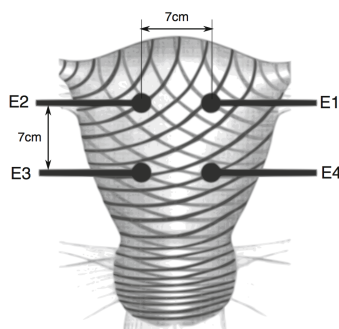
Podatkovna baza TPEHG [6, 9] je zbirka EHG posnetkov zajetih v Univerzitetnem kliničnem centru v Ljubljani na oddelku za ginekologijo. Zajeli so 1211 posnetkov EHG, vendar so jih zaradi različnih napak morali del izločiti. Prav tako so izločili vse posnetke nosečnosti katerih porod je bil spodbujen ali pa se je končal s carskim rezom. Z izločitvijo neprimernih posnetkov tako podatkovna baza TPEHG DB vsebuje skupno 300 posnetkov, za vsako nosečnost po en posnetek. Množica posnetkov se deli na:

- 262 posnetkov zajetih, kjer je nosečnost trajala 37 tednov ali dlje, od tega:
 - 143 posnetkov zajetih pred 26. tednom nosečnosti in
 - 119 posnetkov zajetih v ali po 26. tednu.
- 38 posnetkov zajetih, kjer je šlo za prezgodnji porod, torej kjer je nosečnost trajala manj kot 37 tednov, od tega:

- 19 posnetkov zajetih pred 26. tednom in
- 19 posnetkov zajetih v ali po 26. tednu.

Vsak posnetek je dolg 30 minut katerega tvorijo trije signali. Signali so izračunani iz razlike električnega potenciala elektrod, ki so nameščene na trebuh (Slika 2.2). Razlike potencialov so izračunane po naslednjih formulah:

- $S1 = E2 - E1$
- $S2 = E2 - E3$
- $S3 = E4 - E3$



Slika 2.2: Postavitev elektrod na trebuhu

Pred vzorčenjem je bil vsak signal filtriran z analognim Butterworthovim filtrom mejnih frekvenc $0Hz$ in $5Hz$. Uporabljena frekvenca vzorčenja je bila $20Hz$. Poleg osnovno zajetih posnetkov so v podatkovno bazo dodali tudi vse tri kanale vsakega posnetka filtrirane z digitalnimi Butterworthovimi filtri mejnih frekvenc $0.08Hz$ in $4Hz$, $0.3Hz$ in $4Hz$ ter $0.3Hz$ in $3Hz$.

Vsi posnetki znotraj podatkovne baze TPEHG DB so zapisani v formatu WFDB. Vsak posnetek je sestavljen iz binarne datoteke s končnico *.dat*, ki vsebuje vrednosti posameznih signalov in zaglavne datoteke s končnico *.hea*. Slika 2.3 predstavlja primer zaglavne datoteke. V zaglavni datoteki se nahajajo informacije o posameznih signalih in komentarji, ki vsebujejo dodatno klinično informacijo o nosečnici.

```

tpehg1298 12 20.000110 36000
tpehg1298.dat 16 13107/mV 16 0 0 0 0 1
tpehg1298.dat 16 13107/mV 16 0 0 0 0 1_DOCFILT-4-0.08-4
tpehg1298.dat 16 13107/mV 16 0 0 0 0 1_DOCFILT-4-0.3-3
tpehg1298.dat 16 13107/mV 16 0 0 0 0 1_DOCFILT-4-0.3-4
tpehg1298.dat 16 13107/mV 16 0 0 0 0 2
tpehg1298.dat 16 13107/mV 16 0 0 0 0 2_DOCFILT-4-0.08-4
tpehg1298.dat 16 13107/mV 16 0 0 0 0 2_DOCFILT-4-0.3-3
tpehg1298.dat 16 13107/mV 16 0 0 0 0 2_DOCFILT-4-0.3-4
tpehg1298.dat 16 13107/mV 16 0 0 0 0 3
tpehg1298.dat 16 13107/mV 16 0 0 0 0 3_DOCFILT-4-0.08-4
tpehg1298.dat 16 13107/mV 16 0 0 0 0 3_DOCFILT-4-0.3-3
tpehg1298.dat 16 13107/mV 16 0 0 0 0 3_DOCFILT-4-0.3-4
#Comments:
# RecID 1298
# Gestation 35.9
# Rectime 31.3
# Age 27
# Parity 0
# Abortions None
# Weight None
# Hypertension None
# Diabetes None
# Placental_position None
# Bleeding_first_trimester None
# Bleeding_second_trimester None
# Funneling None
# Smoker None

```

Slika 2.3: Primer zaglavne datoteke

2.4 Gnuplot

Gnuplot je prosto dostopno ukazno vrstično orodje za izrisovanje grafov [14]. Na voljo je za vse popularne operacijske sisteme kot so npr. Windows, Mac OS X ter Linux. Gnuplot nam omogoča interaktivno vizualizacijo matematičnih funkcij in vrst podatkov.

2.5 Java DSP collection

Java DSP collection je zbirka javanskih razredov namenjenih za digitalno procesiranje signalov. Omogoča nam načrtovanje digitalnih filtrov kot so Butterworthovi filtri, Chebyshevi filtri in Besselovi filtri. Poleg razredov za načrtovanje filtrov vsebuje tudi razrede za delo s polinomi, kompleksnimi števili in metode za diskretno Fourierjevo transformacijo. Zbirka je odprtokodna in prosto dostopna na spletu [15].

Poglavje 3

Opis metod

3.1 Principi načrtovanja uporabniških vmesnikov

Pri načrtovanju uporabniškega vmesnika smo sledili sprejetim principom načrtovanja uporabniških vmesnikov. Sledili smo tako osnovnim Mandelovim principom načrtovanja uporabniških vmesnikov [16], ki so:

- Zagotovi nadzor uporabnika
- Reduciraj obremenitev uporabnikovega spomina
- Zagotovi konsistentnost

in bolj definiranim Nielsonovim principom [17]. Tako smo zagotovili, da je uporabniški vmesnik učinkovit in enostaven za uporabo.

3.1.1 Nielsenovi principi

V nadaljevanju je podrobneje opisano 10 Nielsenovih principov za načrtovanje uporabniških vmesnikov.

- Vidljivost statusa sistema

Uporabnik naj bo stalno obveščen o statusu sistema.

- Sistem naj govori jezik uporabnika

Znotraj uporabniškega vmesnika naj se uporabljajo splošne besede in ne tehnični žargon. Izkoristimo lahko morebitne metafore pri katerih moramo biti pazljivi pri izbiri saj lahko hitro zavedejo.

- Uporabnikov nadzor in svoboda

Uporabnik naj ne bo ujetnik znotraj uporabniškega vmesnika. Vsak dialog mora imeti izbiro za preklic. Znotraj uporabniškega vmesnika naj se zagotovi tudi razveljavitev akcij. Dolge operacije naj bodo prekinljive.

- Konsistentnost in standardi

Upoštevati je potrebno princip najmanjšega presenečenja. Podobne stvari naj izgledajo in se obnašajo podobno. Različne stvari naj izgledajo drugačno. Upoštevati je potrebno tudi navodila različnih platform.

- Preprečevanje napak

Vnosna polja naj nadomestijo kombinirani izvlečni sezname v kolikor imamo omejeno možnosti za vnos. Onemogočiti je potrebno napačne oziroma nemogoče akcije v danem trenutku.

- Raje prepoznav kot si zapomni

Prednost naj imajo menuji in ne ukazni jezik. Uporabljajo naj se generični ukazi, kjer je to mogoče. Kjer je potrebno, naj se prikazujejo vse potrebne informacije.

- Fleksibilnost in učinkovitost

Za najbolj pogoste operacije je potrebno zagotoviti bližnjice preko tipkovnice. Uporabljajo naj se ukazne datoteke za morebitne operacije, ki jih je mogoče avtomatizirati.

- Estetika in minimalizem

Uporabniški vmesnik naj bo načrtovan po načelu "manj je več". Izo-
gibati se je potrebno nepotrebnim informacijam in grafikam.

- Javljanje napak, diagnoza in reševanje

Uporabnik naj dobi dobro obvestilo o napaki. Naj bo natančno sezna-
njen zakaj je prišlo do napake. Tehnični detajli naj bodo prikazani na
željo uporabnika. Potrebno je predlagati morebitne rešitve pri poja-
vljenih napakah.

- Pomoč in dokumentacija

Uporabniški vmesnik naj bo v prvi vrsti enostaven za uporabo. Poleg
tega mora biti prisotna tudi dokumentacija za morebitno iskanje po
konkretnih odgovorih.

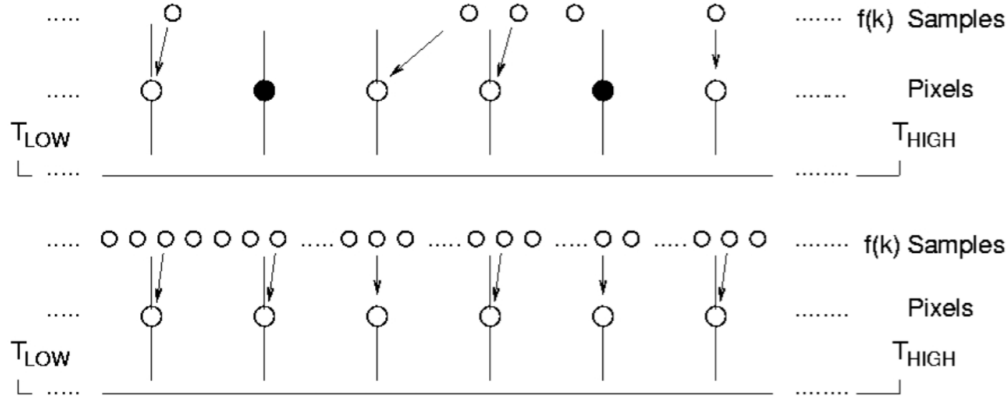
3.2 Vizualizacija signalov

Posnetki EHG nad katerimi smo razvijali uporabniški vmesnik so dolgi 30
minut. To pomeni, da pri frekvenci vzorčenja 20 Hz vsak signal vsebuje
36000 vzorcev. Če želimo prikazati celoten signal, smo pri prikazu omejeni
na širino zaslona na katerem prikazujemo signal. Za izris signalov smo se
poslužili teorije upoštevanja vzorca najbližjega piksla (slika 3.1) [18]. Poleg
tega smo upoštevali tudi prosto izbiro časovne in amplitudne skale. Za vsak
vzorec v signalu $x(i)$, smo koordinate za njegov piksel izračunali po naslednjih
formulah:

$$p_x(i) = \left\lceil \frac{n}{i_{min} - i_{max}} \right\rceil * (i - i_{min}) \quad (3.1)$$

za koordinato x , kjer je n širina risalne površine v piksljih, i_{min} in i_{max} pa
predstavljata začetni in končni indeks intervala vzorcev, ki jih želimo izrisati.
Po formuli:

$$p_y(i) = \frac{x(i)}{C_{adc} * A_s} * n_{ppu} \quad (3.2)$$



Slika 3.1: Teorija upoštevanja vzorca najbližjega piksla

izračunamo koordinato y za predstavljen piksel. Vrednost $x(i)$ predstavlja vrednost vzorca, C_{adc} predstavlja vrednost kalibracije prebranega posnetka, A_s izbrana amplituda v mV , ki predstavlja eno enoto in n_{ppu} kot število pikslov na enoto.

Signal nato izrišemo z vektorskim risanjem črt med izračunanimi piksli. Pri premikanju po osi x za naslednjo povezavo med dvema piksloma upoštevamo pogoj $p_x(i) > p_x(i - 1)$.

3.3 Filtriranje signalov

Zaradi prisotnosti različnih шумov, je filtriranje signalov bistvenega pomena. Na EHG posnetku so lahko šumi prisotni zaradi več razlogov. Prisotni šumi se razlikujejo po prisotnosti v določenih frekvencah. Pri nižjih frekvencah so tipično prisotni šumi zaradi dihanja in raztezanja kože, medtem ko pri visokih frekvencah se šumi pojavljajo zaradi preostalih delov telesa in okolja [19].

Pri filtriranju smo omejeni na dve vrsti filtrov in sicer:

- filtri s končnim impulznim odzivom (FIR)

- filtri z neskončnim impulznim odzivom (IIR)

Rezultat pri filtrih FIR je odvisen samo od preteklih vhodnih vrednosti in trenutne vhodne vrednosti. Zaradi neodvisnosti od preteklih izhodnih vrednosti jim pravimo tudi nerekurzivni filtri. Medtem, ko je pri filtrih IIR rezultat poleg preteklih in trenutnih vhodnih vrednosti odvisen tudi od preteklih izhodnih vrednosti. Prednost filtrov FIR je, da ne povzročijo faznega zamika, kateri je lahko prisoten pri filtrih IIR. Slabost filtrov FIR pa je, da pri isti kvaliteti, ki jo lahko dosežemo s filtri IIR bistveno poveča število računskih operacij. Računska zahtevnost pri filtriranju je pomembna predvsem, če želimo signale filtrirati v realnem času, vendar se moramo zavedati, da je lahko prisoten fazni zamik. V našem primeru signalov sicer ne filtriramo v realnem času, bi pa vseeno radi zagotovili hitro izračunljivost zato smo izbrali filtre IIR.

Natančneje, uporabili smo digitalne Butterworthove filtre. Prednost Butterworthovimi pred ostalimi je njihova monotona prenosna karakteristika. Fazni zamik, ki nastane pri uporabi IIR filtrov pa smo izničili z uporabo dvosmerne sheme filtriranja.

3.3.1 Butterworthovi filtri

Za Butterworthove filtre velja, da je njihova prenosna karakteristika monotona tako v zapornem, kot tudi v prepustnem pasu. Prenosna funkcija nizko prepustnega Butterworthovega filtra je opisana z enačbo:

$$|H(s)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2N}} \quad (3.3)$$

Za prenosno funkcijo visoko prepustnega filtra velja enačba:

$$|H(s)|^2 = \frac{1}{1 + \left(\frac{\omega_c}{\omega}\right)^{2N}} \quad (3.4)$$

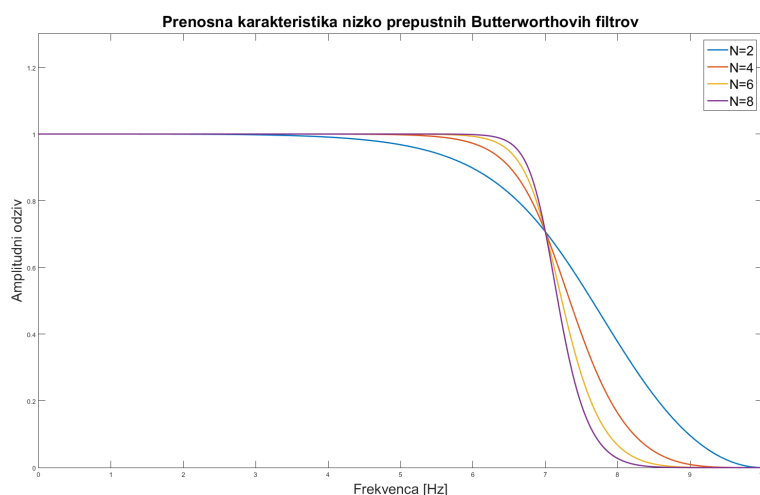
kjer je ω_c mejna frekvenca, N pa red filtra.

Na slikah 3.2 in 3.3 lahko opazujemo prenosno karakteristiko nizko in visoko prepustnih Butterworthovih filtrov različnih redov. V tem primeru je

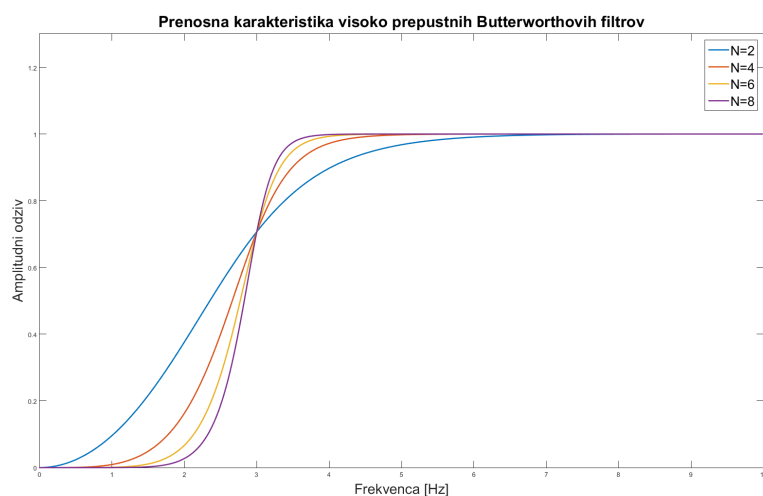
mejna frekvenca nizko prepustnih filtrov 7Hz , visoko prepustnih filtrov pa 3Hz .

V obeh primerih je v prvi vrsti moč opaziti monotonost karakteristike tako v prepustnem kot zapornem pasu. Iz slik je razvidno tudi kako se s povečevanjem reda filtra vedno bolj približujemo stopničasti funkciji oziroma idealnem filtru. Idealno bi veljalo, da nizko prepustni filter, frekvence do izbrane mejne frekvence popolnoma prepušča, višje frekvence pa popolnoma zaduši. Za idealen visoko prepustni filter bi veljalo, da frekvence do mejne frekvence popolnoma duši, medtem ko višje frekvence popolnoma prepušča. Potrebno pa se je zavedati, da s povečevanjem reda filtra povečujemo tudi število računskih operacij, zato mora biti izbira reda filtra primerna željeni hitrosti filtriranja.

Kot smo že omenili je pri Butterworthovih filtrih prisoten fazni zamik. Fazni zamik je posebej problematičen pri visoko prepustnih filtrih. Ker signalov ne filtriramo v realnem času, lahko fazni zamik izničimo z dvosmerno shemo filtriranja.



Slika 3.2: Prenosna karakteristika nizko prepustnih Butterworthovih filtrov različnih redov



Slika 3.3: Prenosna karakteristika visoko prepustnih Butterworthovih filtrov različnih redov

3.3.2 Dvosmerna shema filtriranja

Dvosmerno shemo filtriranja uporabimo za odpravo faznega zamika. Če osnovno filtriranje predstavimo z enačbo:

$$y_1(n) = h(n) * x(n) \quad (3.5)$$

kjer dobimo filtriran signal $y_1(n)$ kot rezultat konvolucije filtra $h(n)$ in osnovnega signala $x(n)$. Dvosmerno filtriranje pa je definirano kot:

$$y_2(n) = h(n) * y_1(-n) \quad (3.6)$$

kjer za naš končni želeni rezultat, torej filtriran signal z dvosmerno shemo filtriranja velja:

$$y(n) = y_2(-n) \quad (3.7)$$

Z uporabo dvosmerne sheme filtriranja pri končnem rezultatu dosežemo tudi prenosno karakteristiko, kot bi jo dobili, če bi izbrali dvakrat večji red filtra.

3.4 Manipulacija posnetkov

Za učinkovito delo s posnetki je potrebna tudi osnovna manipulacija. Najbolj osnovni del manipulacije predstavlja postavljanje oznak in pregledovanje vrednosti v izbranem trenutku. Ob vsakem levem miškinem kliku na risalni površini na mesto klika narišemo vertikalno črto čez celotno risalno površino, ki predstavlja marker. Na markirano mesto lahko preko gradnikov dostopnih v orodni vrstici postavljamo statične oznake.

Posamezna oznaka je programsko predstavljena kot samostojen objekt, ki vsebuje atributa pozicijski indeks *positionIndex* in tip oznake. Vsako postavljeno oznako na risalni površini predstavimo kot vektorski izris črte med točkama $(positionIndex, 0)$ in $(positionIndex, height)$, kjer *height* predstavlja višino risalne površine.

Posebni oznaki sta oznaki za označevanje lokalnega intervala. S pomočjo označenega lokalnega intervala lahko opravljamo tudi druge manipulacije. Možnost povečevanja označenega dela posnetka, ki je dostopna iz orodne vrstice smo realizirali tako, da začetek lokalnega intervala predstavlja začetek na risalni površini in konec lokalnega intervala konec na risalni površini. S povečevanjem označenega dela smo v primerih, kjer želimo določen del signala prikazati na celotni risalni površini dosegli neodvisnost od izbrane časovne skale.

Označen lokalni interval lahko uporabimo tudi za izrez določenega dela posnetka. Želeni del posnetka ob uspešnem izrezu shranimo v novo datoteko v formatu WFDB. Tako lahko izrezan del posnetka interpretiramo kot nov posnetek, katerega lahko prav tako uporabimo v našem uporabniškem vmesniku za nadaljnjo delo.

3.5 Fourierova transformacija

Fourierova transformacija je matematičen pristop, ki nam omogoča transformacijo funkcije v časovnem prostoru v njen frekvenčni prostor. V frekvenčnem prostoru lahko opazujemo, kako so porazdeljene določene frekvence v

opazovanem signalu. Matematično je definirana kot:

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-2\pi ift} dt \quad (3.8)$$

kjer je $x(t)$ zvezen signal v časovnem prostoru.

Ker so signali, ki jih želimo transformirati predstavljeni diskretno, za tovrstne transformacije uporabljamo diskretno Fourierjevo transformacijo (angl. Discrete Fourier transformation - DFT). Za DFT velja izpeljava:

$$X(m) = \sum_{n=0}^{N-1} x(n)e^{-2\pi inm/N} \quad (3.9)$$

kjer je $x(n)$ diskreten signal v časovnem prostoru in $X(m)$ iskana transformacija predstavljena v frekvenčnem prostoru.

Pri DFT tako izvajamo transformacijo diskretnega signala predstavljenega v časovnem prostoru z dolžino N , vzorčenega s frekvenco vzorčenja f_s . Kot rezultat pa dobimo prisotnost analitičnih frekvenc $f_{an}(m)$, ki so definirane kot:

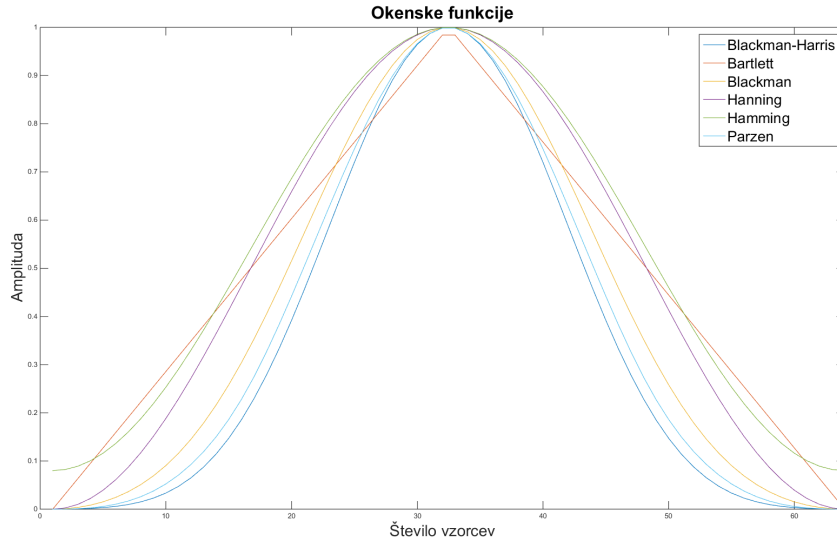
$$f_{an}(m) = \frac{mf_s}{N}, \text{ za } m = 0, 1, 2, \dots, N-1 \quad (3.10)$$

V primeru, ko so v signalu prisotne frekvence, ki ne ustrezajo enačbi 3.10 govorimo o puščanju. To pomeni, da vmesne frekvence denimo $1.5f_s/N$ uhajajo v prostor drugih frekvenčnih komponent. Puščanje določenih frekvenc v prostor ostalih lahko zavaja pri analizi frekvenčnega prostora. Tega pojava ne moremo popolnoma preprečiti lahko pa zmanjšamo učinek puščanja z uporabo okenskih funkcij [20].

Ker je DFT računsko zahtevno, njegova časovna zahtevnost je $\mathcal{O}(n^2)$, se za izračun tovrstnih transformacij uporablja algoritem hitre Fourierjeve transformacije (angl. Fast Fourier transformation - FFT) z računsko zahtevnostjo $\mathcal{O}(n \log n)$

3.6 Okenske funkcije

Z uporabo okenskih funkcij lahko zmanjšamo učinek puščanja, ki lahko nastopi pri transformaciji iz časovnega v frekvenčni prostor. Okensko funkcijo



Slika 3.4: Karakteristike različnih okenskih funkcij

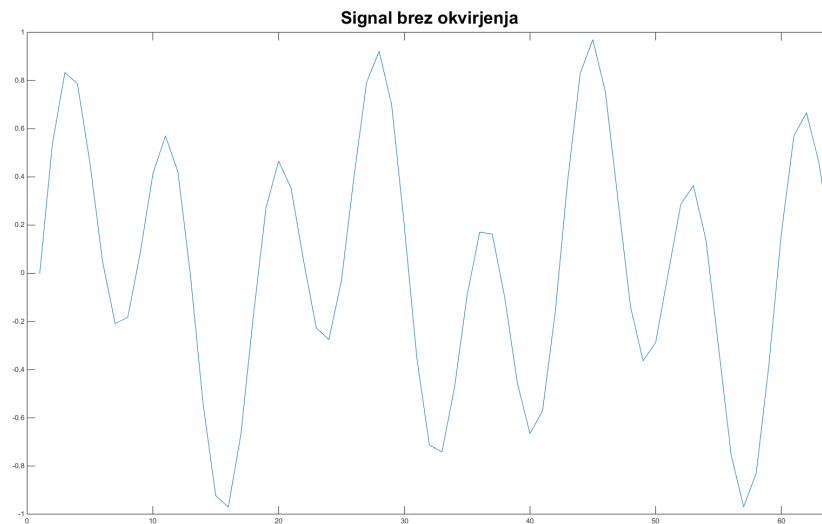
uporabimo tako, da diskretni signal predstavljen v časovnem prostoru dolžine N zmnožimo z želeno okensko funkcijo. Matematično lahko DFT z uporabo okenske funkcije zapišemo kot:

$$X(m) = \sum_{n=0}^{N-1} \omega(n)x(n)e^{-2\pi inm/N} \quad (3.11)$$

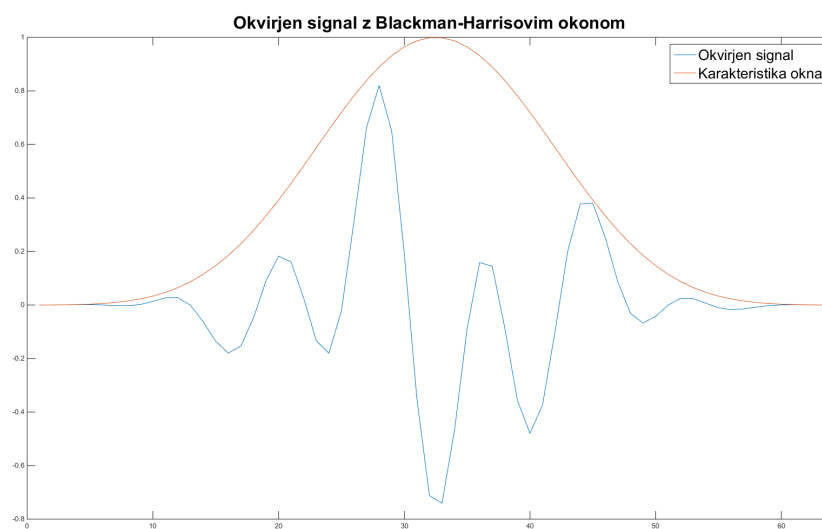
kjer je $\omega(n)$ okenska funkcija.

Poznamo več vrst okenskih funkcij, katere uporabimo odvisno od namena oziroma želenega rezultata. Na sliki 3.4 so prikazane karakteristike nekaterih obstoječih okenskih funkcij.

Sliki 3.5 in 3.6 predstavljata uporabo okenske funkcije. Na sliki 3.5 je vzorčen signal nad katerim želimo uporabiti okensko funkcijo. V našem primeru smo uporabili okensko funkcijo *Blackman – Harris*, katere karakteristika je tudi na sliki 3.4. Rezultat pri uporabi okenske funkcije je predstavljen na sliki 3.6, na kateri je za lažjo predstavo tudi obris karakteristike uporabljene funkcije.



Slika 3.5: Vzorčni signal pred uporabo okenske funkcije

Slika 3.6: Vzorčni signal po uporabi okenske funkcije *Blackman – Harris*

3.7 Amplitudni in močnostni spekter

Amplitudni spekter uporabljamo za grafično predstavitev porazdeljenosti posameznih frekvenc in njihovih magnitud znotraj opazovanega signala. Za prikaz amplitudnega spektra je najprej potrebna transformacija iz časovnega prostora v frekvenčni prostor. V poglavju 3.5 smo predstavili pristop DFT, ki je namenjen tovrstni transformaciji. Če je $X(m)$ predstavljen signal v frekvenčnem prostoru lahko njegov amplitudni spekter $A(m)$ izračunamo kot:

$$A(m) = \frac{|X(m)|}{N} \quad (3.12)$$

Ker je frekvenčni prostor signala $X(m)$ predstavljen v kompleksni ravnini za njegovo absolutno vrednost $|X(m)|$ velja:

$$|X(m)| = \sqrt{a^2 + b^2} \quad (3.13)$$

kjer je a realna in b imaginarna komponenta kompleksnega števila.

Za grafično predstavitev moči posameznih frekvenc uporabljamo tako imenovane močnostne spektre. Iz signala predstavljenega v frekvenčnem prostoru lahko močnostni spekter izračunamo po naslednji enačbi:

$$P(m) = \frac{|X(m)|^2}{N} \quad (3.14)$$

kjer za absolutno vrednost $|X(m)|$ ponovno velja enačba 3.13.

3.8 Avtoregresivna metoda za ocenitev spektra

Avtoregresija (AR) je ena izmed metod za ocenjevanje močnostnih spektrov. Omogoča nam bolj zvezno predstavitev poteka močnostnega spektra. AR model napove trenutno vrednost v časovnem zaporedju na podlagi preteklih vrednosti v istem zaporedju. AR model je definiran z naslednjo diferenčno enačbo:

$$x(n) = -a_{1x}(n-1) - \dots - a_{px}(n-p) + v(n) \quad (3.15)$$

kjer so $a_1 \dots a_p$ fiksni parametri modela in $v(n)$ prisoten kot beli šum z varianco σ_v^2 definirano kot:

$$E[v^2(n)] = \sigma_v^2 \quad (3.16)$$

Izhodni vzorec $x(n)$ je sestavljen kot linearna kombinacija p predhodnih izhodnih vzorcev $x(n-1) \dots x(n-p)$ in trenutna vrednost $v(n)$. Številu p rečemo tudi red modela. Čeprav za red modela lahko izberemo katero koli število manjše od števila vzorcev, naj red modela ne bi presegal kvadratnega korena števila vzorcev. Pri previsoko izbranem redu modela je lahko računanje občutno počasnejše tudi od FFT. Priporočljiva uporaba avtoregresivne metode je, da najprej uporabimo FFT s katerim določimo število prisotnih vrhov. Odčitano število vrhov pomnožimo z nizkim faktorjem in dobljen rezultat uporabimo kot red modela [12].

Z uporabo Z-transformacije lahko AR model enakovredno predstavimo kot prenosno funkcijo $H(z)$ za katero velja enačba:

$$H(z) = \frac{1}{1 + a_1 z^{-1} + \dots + a_p z^{-p}} \quad (3.17)$$

Ko imamo ocenjene parametre modela $a_1 \dots a_p$ in σ_v^2 iz opazovanega signala lahko izračunamo močnostni spekter $S_x(z)$ AR modela po naslednji enačbi:

$$S_x(z) = H(z)H(z^{-1})\sigma_v^2 \quad (3.18)$$

oziroma, če $H(z)$ dodatno izpeljemo in z predstavlja $z = e^{j\omega}$:

$$S_x(e^{j\omega}) = \frac{\sigma_v^2}{|1 + \sum_{k=1}^p a_k e^{-j\omega k}|^2} \quad (3.19)$$

Iz vhodnih podatkov lahko parametre $a_1 \dots a_p$ ocenimo z uporabo [21]:

- Avtokorelacijske ali kovariančne metode
- Spremenjene kovariančne metode
- Burgove metode

3.9 Analiza močnostnih spektrov

Pri analizi signalov smo se osredotočili na linearne tehnike. Pri vsakem izračunu lahko poljubno izbiramo frekvenčno območje v katerem želimo ovrednotiti izračunane spektre. Tako poleg izračunanih spektrov, izračunamo tudi:

- vrh spektra po enačbi:

$$p_{max} = \max_{i=i_{low}}^{i=i_{high}} P(i) \quad (3.20)$$

kjer i_{min} in i_{high} predstavljata indeksa meje intervala za analizo.

- frekvenco vrha spektra po enačbi:

$$f_{max} = \frac{f_s}{N} \arg(\max_{i=i_{low}}^{i=i_{high}} P(i)) \quad (3.21)$$

kjer f_s predstavlja frekvenco vzorčenja in N število vzorcev v našem diskretnem signalu.

- mediano frekvence po enačbi:

$$f_{med} = i_m \frac{f_s}{N}, \quad \sum_{i=i_{low}}^{i=i_m} P(i) \approx \sum_{i=i_{m+1}}^{i=i_{high}} P(i) \quad (3.22)$$

kjer i_m predstavlja indeks na mestu, ki predstavlja mediano frekvence.

3.10 Razvoj grafičnega vmesnika EHGLab

Razvoj grafičnega uporabniškega vmesnika EHGLab je potekal v okolju Net-Beans IDE verzije 8.02 v programskem jeziku Java verzije 1.8. Z izbiro programskega jezika Java smo dosegli neodvisnost od uporabnikovega operacijskega sistema. Za grafično načrtovanje smo uporabili grafične gradnike Swing, kateri so sestavni del programskega jezika Java.

Uporabniški vmesnik je sestavljen iz naslednjih komponent:

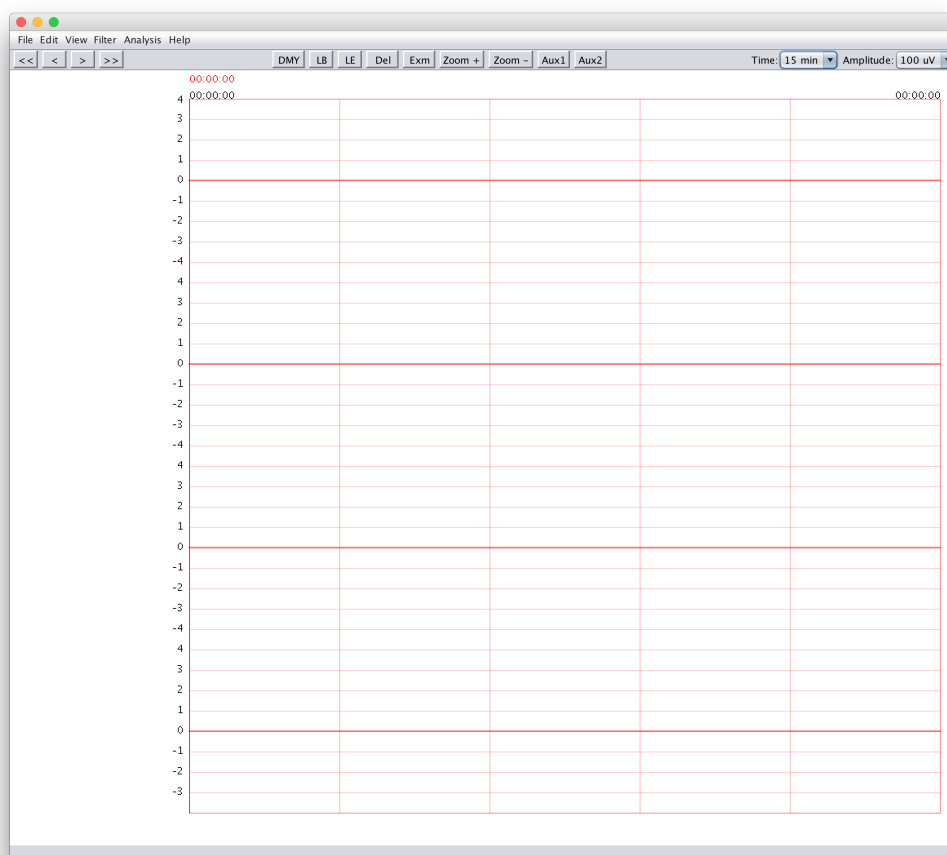
- Glavno okno
- Vrstični menu
- Orodna vrstica
- Prostor za podrobnosti o signalih
- Risalna površina
- Statusna vrstica

3.10.1 Glavno okno

Predstavlja glavno komponento uporabniškega vmesnika, ki se odpre ob zagonu. Vsebuje vse ostale komponente, ki so potrebne za nadaljnjo delo. Slika 3.7 prikazuje zaslonski posnetek glavnega okna takoj ob zagonu uporabniškega vmesnika.

3.10.2 Vrstični menu

V vrstičnem meniju (slika 3.8), se nahajajo glavne funkcionalnosti uporabniškega vmesnika. Vrstični menu sledi zgledu modela FEVH (*File*, *Edit*, *View*, *Help*), poleg naštetih opcij smo dodali tudi opciji *Filter* in *Analysis*.



Slika 3.7: Glavno okno uporabniškega vmesnika EHGLab

Menu File

Znotraj menuja *File* se nahajajo končne izbire za branje, shranjevanje posnetkov, ustvarjanje zaslonskega posnetka in izhod iz uporabniškega vmesnika. V nadaljevanju so opisane končne izbire znotraj menuja *File*.

Open Končna izbira *Open* je namenjena za branje poljubnega posnetka v formatu WFDB. Uporabniku ponudimo dialog za izbiro posnetka. V izogib napakam, za izbiro dopuščamo samo zaglavne datoteke (končnica *.hea*), kjer so zapisani meta podatki za posamezen posnetek. Ko uporabnik potrdi iz-



Slika 3.8: Izgled vrstičnega vrstičnega menuja

bran signal, iz zaglavne datoteke preberemo pot do binarne datoteke kjer se nahajajo podatki o posnetku. Ob uspešnem branju podatke izrišemo na risalno površino.

Save Ker lahko znotraj uporabniškega vmesnika opravljamo različne manipulacije s posnetkom smo dodali tudi končno izbiro *Save*, ki je namenjena shranjevanju posnetkov. Pri izbiri uporabniku ponudimo dialog za shranjevanje in predlagamo novo ime, tako da trenutno ime posnetka ustrezno podaljšamo. Posnetke shranjujemo v formatu WFDB.

Save as Pod končno izbiro *Save as* uporabniku dopuščamo izbiro formata za shranjevanje. Poleg standardnega formata WFDB lahko podatke o prikazanem posnetku shranimo tudi v tekstovno datoteko.

Možnost shranjevanja v tekstovno datoteko je koristno predvsem, če želimo podatke uporabljati tudi v drugih programih, ki ne podpirajo branja iz formata WFDB.

Dump Končna izbira *Dump* ustvari zaslonski posnetek trenutnega stanja v uporabniškem vmesniku. Zaslonski posnetek se shrani v isti imenik, kjer se nahaja trenutno prebran posnetek.

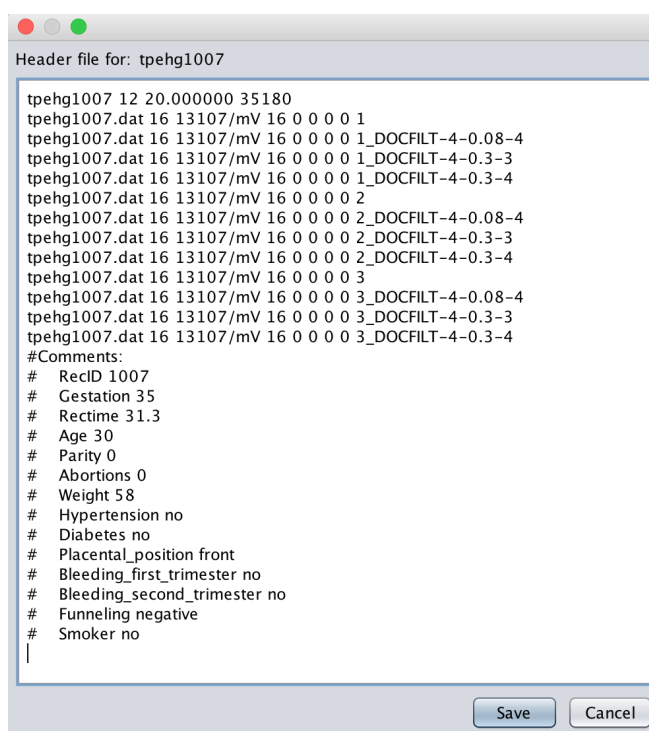
Quit Izhod iz uporabniškega vmesnika.

Menu Edit

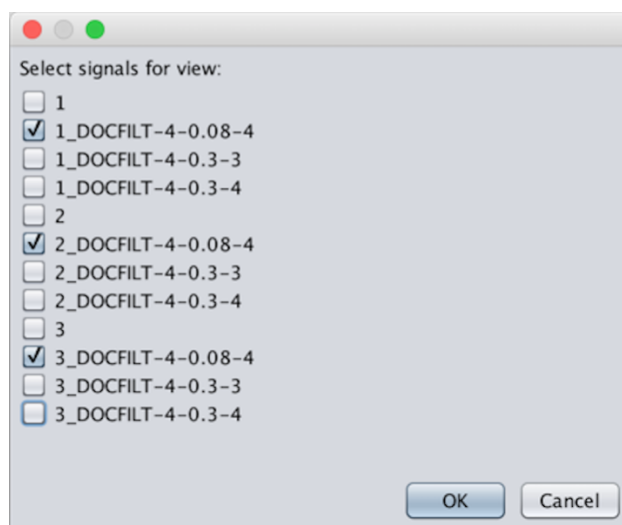
Znotraj menuja *Edit* se nahajajo v nadaljevanju našete izbire.

Snip Končna izbira *Snip* je namenjena rezanju posnetka. Omejitev pri tej izbiri je, da morajo biti meje izreza označene z lokalnim intervalom. V kolikor lokalni interval ni označen, uporabnik dobi opozorilo o ne obstoju le tega. V kolikor je akcija uspešna, se izrezan posnetek shrani v isti imenik, kjer se nahaja trenutno prebran posnetek. Ime se ustvari iz trenutnega imena posnetka in dodanega podaljška iz katerega je moč razbrati, da gre za izrez posnetka. Izrezan posnetek je v formatu WFDB, zato se ustvari tudi nova zaglavna datoteka, ki časovno definira meje izreza.

Header Končna izbira *Header* odpre v zunanjem oknu (slika 3.9) možnost pregleda in urejanja zaglavne datoteke odprtega posnetka.



Slika 3.9: Dialog za pregledovanje in urejanje zaglavnih datotek



Slika 3.10: Dialog za izbiro signalov namenjenih za risanje na risalno površino

Menu View

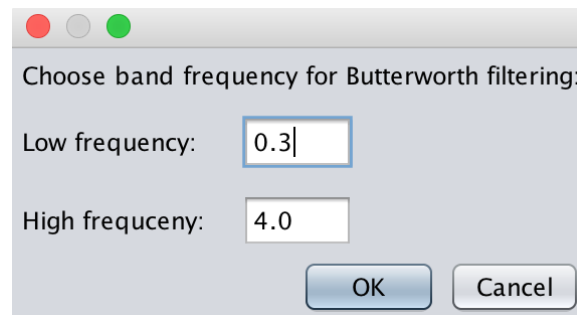
Znotraj menuja *View* se nahaja končna izbira *Signals*. Ker na risalno površino izrisujemo samo štiri signale znotraj posnetka, lahko preko končne izbire *Signals* izbiramo med naborom signalov, katere želimo prikazati na risalni površini. Na sliki 3.10 je prikazan dialog za izbiro signalov. Vsak signal ima potrditveno polje za označevanje zelenih signalov za prikaz in za identifikacijo prirejeno ime iz zaglavne datoteke.

Menu Filter

Menu *Filter* je namenjen filtriranju signalov. V nadaljevanju so opisane posamezne končne izbire.

Filter order Predstavlja dodaten menu s končnimi izbirami za izbiro reda filtra. Red filtra lahko izbiramo med vrednostnimi od 1 do 8. Red filtra je omejen na najvišjo vrednost 8, ker z višjimi redi povečujemo računsko zahtevnost filtrov. Privzeto je red filtra nastavljen na 4.

Custom Butterworth Preko končne izbire *Custom Butterworth* se odpre dialog (slika 3.11) za vnos poljubnih mejnih frekvenc. Privzeto so vrednosti nastavljene na $0.3Hz$ in $4Hz$. Pri vpisu mejnih frekvenc lahko pride do napak, recimo vpisovanje teksta zato ob potrditvi preverimo, če so vrednosti decimalna števila. V kolikor je prišlo do napake uporabnika obvestimo, da so dovoljena le decimalna števila. Po uspešni potrditvi opravimo filtriranje.



Slika 3.11: Dialog za vnos mejnih frekvenc Butterworthovih filtrov

Butterworth [0.08Hz - 4Hz] Hitra izbira za filtriranje z mejnimi frekvencami $0.08Hz$ in $4Hz$.

Butterworth [0.3Hz - 4Hz] Hitra izbira za filtriranje z mejnimi frekvencami $0.3Hz$ in $4Hz$.

Butterworth [0.3Hz - 3Hz] Hitra izbira za filtriranje z mejnimi frekvencami $0.3Hz$ in $3Hz$.

Auxn Dodane so tudi dodatne končne izbire brez funkcionalnosti. Namenjene so morebiti za dodajanje novih filtrov in/ali novih hitrih izbir z določenimi mejnimi frekvencami.

Ker digitalni Butterworthovi filtri povzročijo fazni zamik filtriranega signala pri vseh filtrih uporabimo dvosmerno shemo filtriranja. Tako izničimo

prisoten fazni zamik. Pri vseh končnih izbirah se za red filtra upošteva izbran red iz menuja *Filter order*.

Ob ustrezni izbiri filtriramo samo prikazane signale na risalni površini. Po uspešnem filtriranju, filtriranim posnetkom določimo ime tako, da trenutnemu imenu signala dodamo podaljšek. Vse filtrirane signale dodamo tudi trenutnemu posnetku, ki jih lahko prikažemo na risalni površini ali shranimo v nov posnetek.

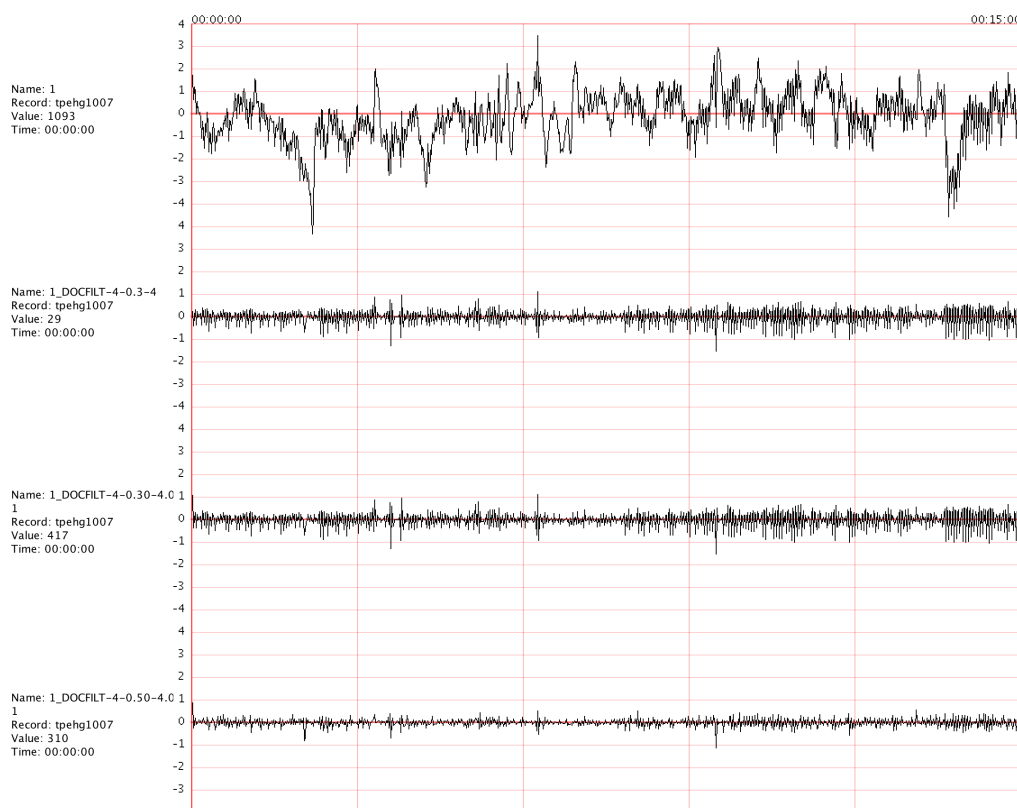
Slika 3.12 prikazuje štiri signale. Prvi signal z imenom 1 je izvorni, ne filtrirani iz podatkovne baze TPEHG. Drugi signal z imenom 1_*DOCFILT* – 4 – 0.3 – 4 je filtriran signal 1 prav tako iz podatkovne baze TPEHG. Tretji signal 1_*DOCFILT* – 4 – 0.30 – 4.00_1 je filtriran signal 1, kjer je bilo filtriranje opravljeno znotraj uporabniškega vmesnika z v naprej pripravljenim filtrom mejnih frekvenc 0.3Hz in 4Hz. Četrty signal 1_*DOCFILT* – 4 – 0.50 – 4.00_1 je prav tako filtriran signal 1, le da smo pri tem signalu izbrali *Custom Butterworth* in določili mejne frekvence 0.5Hz in 4Hz.

Menu Analysis

Menu *Analysis* je namenjen izračunu amplitudnih in močnostnih spektrov in njihove analize. V nadaljevanju so predstavljene končne izbire, ki so na voljo.

Select window Predstavlja dodaten menu s končnimi izbirami za izbiro okenske funkcije uporabljene pri izračunu spektrov. Izbrano okensko funkcijo uporabljamo pri izračunu vseh možnih spektrov. Na voljo so naslednje okenske funkcije Bartlett, Blackman, Blackman-Harris, Hamming, Hanning, Parzen, Rectangular in Welch. Privzeto je izbrana okenska funkcija *Rectangular*, ki pa nima vpliva na rezultat in je enako kot, če okenske funkcije ne bi uporabili.

Normalization area Končna izbira *Normalization area* nam odpre dodaten dialog za vnos mejnih frekvenc pri normalizaciji. Pred potrditvijo vnosa



Slika 3.12: Izvorni signal in trije filtrirani signali

preverimo vpisane vrednosti in morebitno napako opozorimo z opozorilom o neveljavnem vnosu.

Analysis area Končna izbira *Analysis area* nam odpre dodaten dialog za vnos mejnih frekvenc uporabljenih za analizo. Pred potrditvijo vnosa preverimo vpisane vrednosti in na morebitno napako opozorimo z opozorilom o neveljavnem vnosu.

Model order Končna izbira *Model order* nam odpre dodaten dialog za vnos reda modela uporabljenega pri avtoregresivni metodi za oceno spektra. Najvišja in privzeta vrednost za red modela je $\lceil \sqrt{N} \rceil$, kjer N predstavlja število vzorcev v signalu.

Amplitude spectrum Končna izbira *Amplitude spectrum* je namenjena izračunu amplitudnega spektra. Transformacija v frekvenčno predstavitev se opravi z algoritmom FFT, ki je priložen programskemu paketu WFDB. Dobljene vrednosti so predstavljene kot kompleksna števila, zato za predstavitev na grafu uporabimo njihovo absolutno vrednost.

Amplitude spectrum (MEM) Končna izbira *Amplitude spectrum (MEM)* je prav tako namenjena izračunu amplitudnega spektra. V tem primeru frekvenčni spekter ocenimo z avtoregresivno metodo, ki je priložena programskemu paketu WFDB. Pri uporabi avtoregresivne metode za ocenitev spektra upoštevamo red modela, ki je nastavljen pod končno izbiro *Model order*.

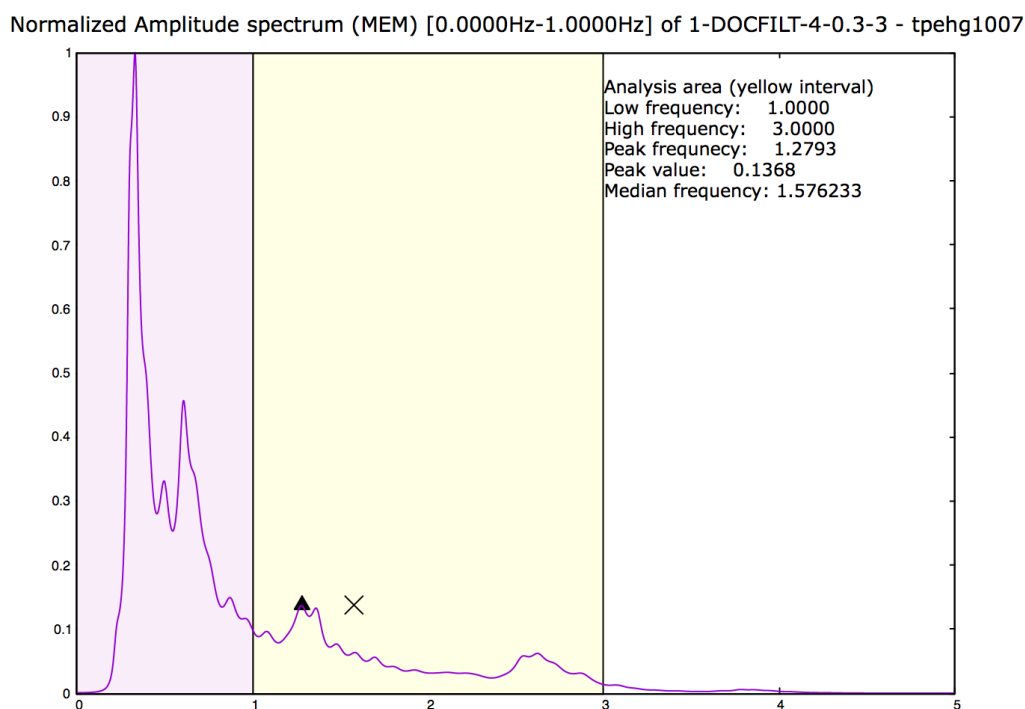
Power spectrum S končno izbiro *Power spectrum* izračunamo močnostni spekter. V tem primeru uporabimo algoritem FFT.

Power spectrum (MEM) S končno izbiro *Power spectrum (MEM)* prav tako izračunamo močnostni spekter. V tem primeru uporabimo avtoregresivno metodo za ocenitev spektra. Pri uporabi avtoregresivne metode za ocenitev spektra upoštevamo red modela, ki je nastavljen pod končno izbiro *Model order*.

Auxn Dodane so tudi dodatne končne izbire brez funkcionalnosti. Namenjene so morebitnim dodajanjem novih metod za oceno spektra in/ali izračunom dodatnih cenilk.

Poleg opisanih končnih izbir, imamo v enakem zaporedju za vsako od naštetih na voljo tudi izračun normaliziranih spektrov. Pri normalizaciji se upošteva območje znotraj intervala mejnih frekvenc določenih pod končno izbiro *Normalization area*.

Izbran spekter znotraj menuja *Analysis* izračunamo na vseh prikazanih signalih na risalni površini. Poleg tega pred izračunom preverimo morebiten



Slika 3.13: Normaliziran frekvenčni spekter - AR metoda za ocenitev spektra

obstoj lokalnega intervala postavljenega z oznakami. V kolikor je mogoče določiti lokalni interval, spektre računamo znotraj le tega. Za vsak izračunan spekter opravimo tudi analizo spektra. Na izračunanem spektru znotraj območja analize izračunamo vrh spektra, frekvenco vrha spektra in mediano frekvence. Vsak izračunan spekter in njegove cenilke predstavimo v zunanjem oknu na grafu. Vrh spektra na grafu označimo s simbolom ▲ mediano frekvence pa s simbolom ×.

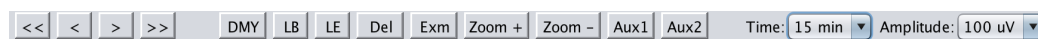
Na grafu tudi barvno označimo območja analize in v kolikor gre za normaliziran spekter tudi območje normalizacije. Slika 3.13 prikazuje normaliziran spekter ocenjen z avtoregresivno metodo. Območje normalizacije je med 0Hz in 1Hz . Na grafu je to območje označeno z vijoličnim ozadjem. Podatki o normalizaciji so vidni tudi v naslovu grafa. Za območje analize smo pri izračunu tega spektra določili območje med 1Hz in 3Hz kateri je označen z rumenim ozadjem.

Menu Help

Znotraj menuja *Help* se nahaja končna izbira s podatki o avtorju in preusmeritev na to diplomsko nalogo.

3.10.3 Orodna vrstica

V orodni vrstici (slika 3.14) se nahajajo orodja za premikanje po risalni površini, postavljanje oznak, povečevanju označenega dela posnetka in pa spreminjanje amplitudne in časovne skale. V nadaljevanju so podrobno opisane vse izbire.



Slika 3.14: Izgled orodne vrstice

Premikanje po risalni površini

S pomočjo izbire časovne skale, lahko samostojno določimo kolikšen del posnetka želimo na risalni površini. V kolikor je posnetek daljši od izbrane časovne skale smo za premikanje po risalni površini dodali gumbe v orodno vrstico. Za premikanje po risalni površini so na voljo naslednji gumbi:

- << kar pomeni premik celotne risalne površine v levo
- < kar pomeni premik polovice risalne površine v levo
- > kar pomeni premik polovice risalne površine v desno
- >> kar pomeni premik celotne risalne površine v desno

Tako lahko s pomočjo gumbov pregledujemo vrednosti celotnega signala neodvisno od izbrane skale.

Postavljanje oznak

Na risalno površino, lahko na poljubna mesta postavljamo tudi določene oznake:

- gumb *DMY* je namenjen oznaki brez vsebine
- gumb *LB* predstavlja oznako za začetek lokalnega intervala
- gumb *LE* predstavlja oznako za konec lokalnega intervala

Oznake se postavljajo na način, da na risalni površini, kjer želimo postaviti določeno oznako kliknemo z levim miškinim klikom in izberemo ustrezno oznako. *DMY* oznak je na risalni površini lahko poljubno mnogo. Ker na posnetku dopuščamo možnost samo enega lokalnega intervala velja omejitev za oznaki *LB* in *LE*, tako da sta lahko prisotni samo enkrat na risalni površini. Postavljeni oznaki *LB* in *LE* že predstavljata lokalni interval v ostalih operacijah in ni potrebe po dodatni akciji.

Upravljanje z oznakami

Poleg gumbov za postavljanje oznak, sta prisotna tudi gumba za upravljanje z oznakami:

- gumb *Del* je namenjen brisanju oznak
- gumb *Exm* je namenjen pregledu vrednosti na mestu postavljenih oznak

Oba gumba lahko uporabimo tako, da najprej na risalni površini z levim miškinim klikom postavimo marker v bližino oznake in prožimo akcijo s klikom na ustrezen gumb. Uporabniški vmesnik sam zazna najbližjo oznako in opravi ustrezno operacijo.

Povečevanje izbranega dela signalov

Za povečevanje in oddaljevanje izbranega dela signalov sta v orodni vrstici na voljo dva gumba:

- *Zoom +* je namenjen povečevanju signalov
- *Zoom -* je namenjen vrnitvi v prvotno stanje glede na časovno skalo

Pogoj za uspešno povečevanju je postavljen lokalni interval. Ob uspešni akciji signal znotraj označenega intervala izrišemo čez celotno površino brez upoštevanja časovne skale.

Izbira časovne skale

Za potrebe bolj podrobnega pregleda določenih delov signala smo omogočili tudi prosto izbiro časovne skale. Pri izbiri časovne skale smo zaradi preprečevanja napak uporabili izvlečno kombiniran seznam, ki je na voljo v orodni vrstici. Za izbiro so na voljo naslednje vrednosti: *30min*, *15min*, *10min*, *5min*, *2min* in *1min*. Privzeto ob zagonu vmesnika je izbrana vrednost *15min*. Ob izbiri vrednosti iz seznama se risalna površina prilagodi izbrani vrednosti.

Izbira amplitudne skale

Zaradi različnih vrednosti amplitud med posameznimi signali smo omogočili tudi prosto izbiro amplitudne skale. Pri izbiri amplitudne skale smo prav tako kot pri izbiri časovne skale uporabili izvlečno kombiniran seznam, ki je na voljo v orodni vrstici. Za izbiro so na voljo naslednje vrednosti: $1\mu V$, $2\mu V$, $5\mu V$, $10\mu V$, $20\mu V$, $50\mu V$, $100\mu V$, $200\mu V$, $500\mu V$, $1mV$, $1mV$, $2mV$, $4mV$, $5mV$, $10mV$, $20mV$. Privzeto ob zagonu vmesnika je izbrana vrednost $100\mu V$. Izbrana amplituda predstavlja eno enoto na risalni površini.

3.10.4 Risalna površina

Risalna površina zavzema največji del uporabniškega vmesnika. Omogoča nam prikaz signalov v časovnem prostoru. Na risalno površino lahko hkrati narišemo največ štiri signale, ki so pozicionirani eden pod drugega.

Slika 3.15 prikazuje celotno risalno površino z izrisanimi štirimi signali. Os Y, ki predstavlja vrednost posameznega signala v določenem trenutku smo

razdelili na enote. Na risalno površino poleg samih signalov izrisujemo tudi morebitne postavljene oznake in marker trenutnega položaja. Modre vertikale predstavljajo postavljene oznake, odebeljena rdeča vertikala pa predstavlja marker trenutnega položaja in čas pri katerem je postavljen marker. Risalna površina se odzove tudi na dogodke miške. Marker trenutnega položaja lahko na risalni površini premikamo z vlečenjem miške ali s klikom na zeleno mesto. Oznake se izrišejo na risalno površino na mesto markerja trenutnega položaja v primeru, če prožimo ustrezno akcijo iz orodne vrstice.

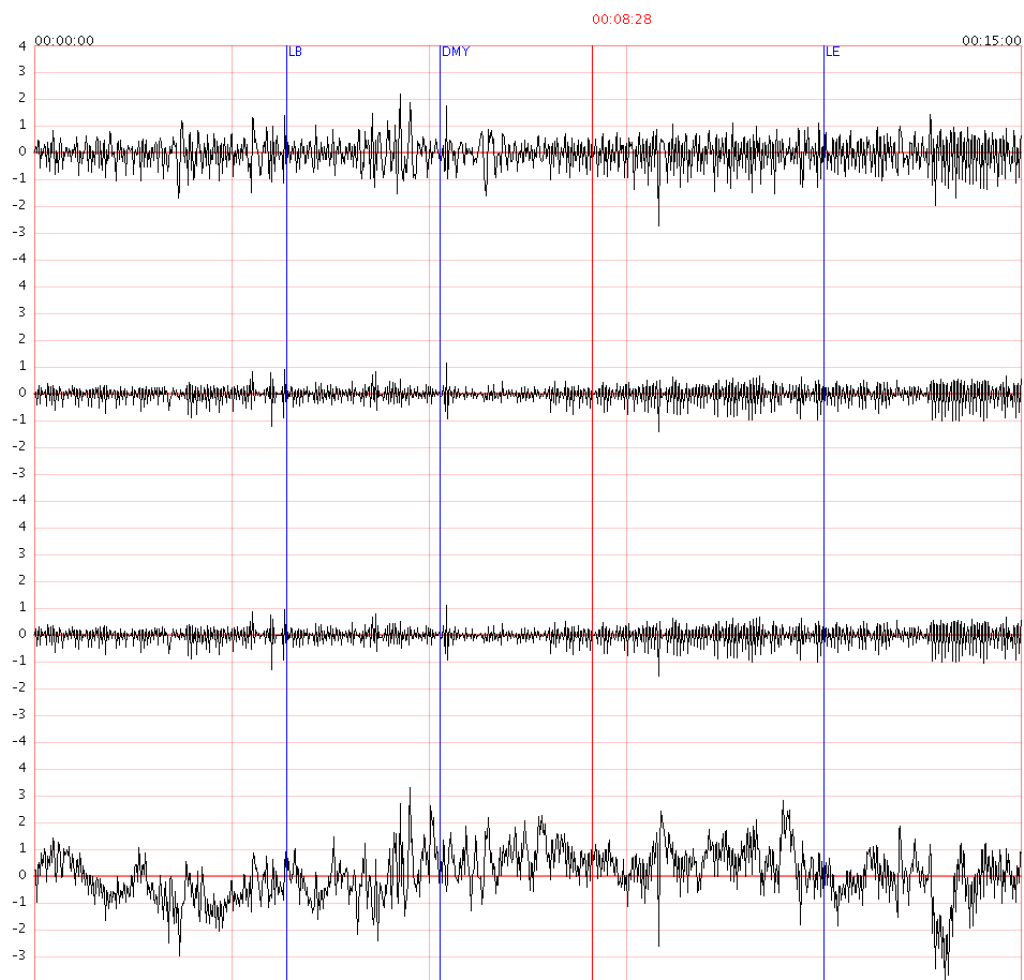
3.10.5 Prostor za podrobnosti o signalih

Levo ob risalni površini se nahaja prostor za podrobnosti o prikazanih signalih. Za vsak prikazan signal izpisujemo naslednje podatke:

- Ime signala, ki je definiran v zaglavni datoteki
- Ime posnetka
- Vrednost pri postavljenem markerju ali opazovani oznaki
- Čas pri postavljenem markerju ali opazovani oznaki

3.10.6 Statusna vrstica

Na dnu uporabniškega vmesnika se nahaja statusna vrstica, ki je namenjena obveščanju uporabnika o trenutnem statusu sistema.



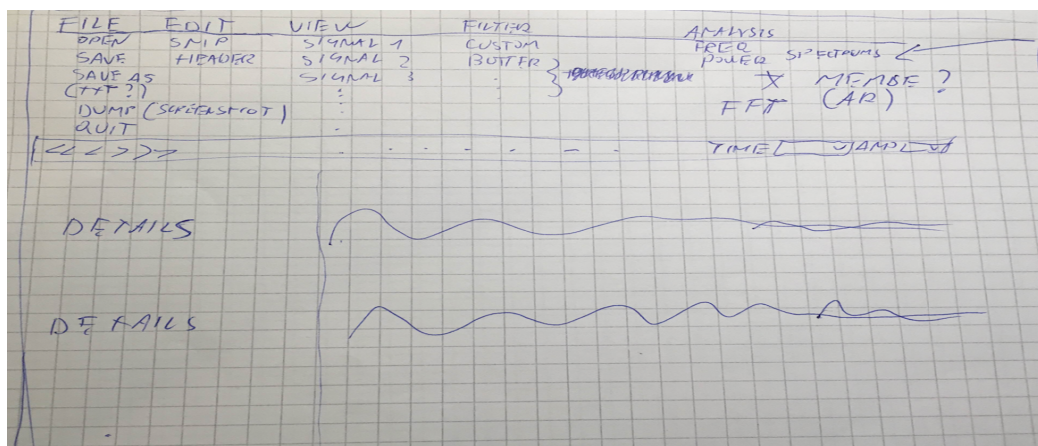
Slika 3.15: Risalna površina z izrisanimi signali

Poglavje 4

Rezultati

4.1 Grafični vmesnik EHGLab in primeri uporabe

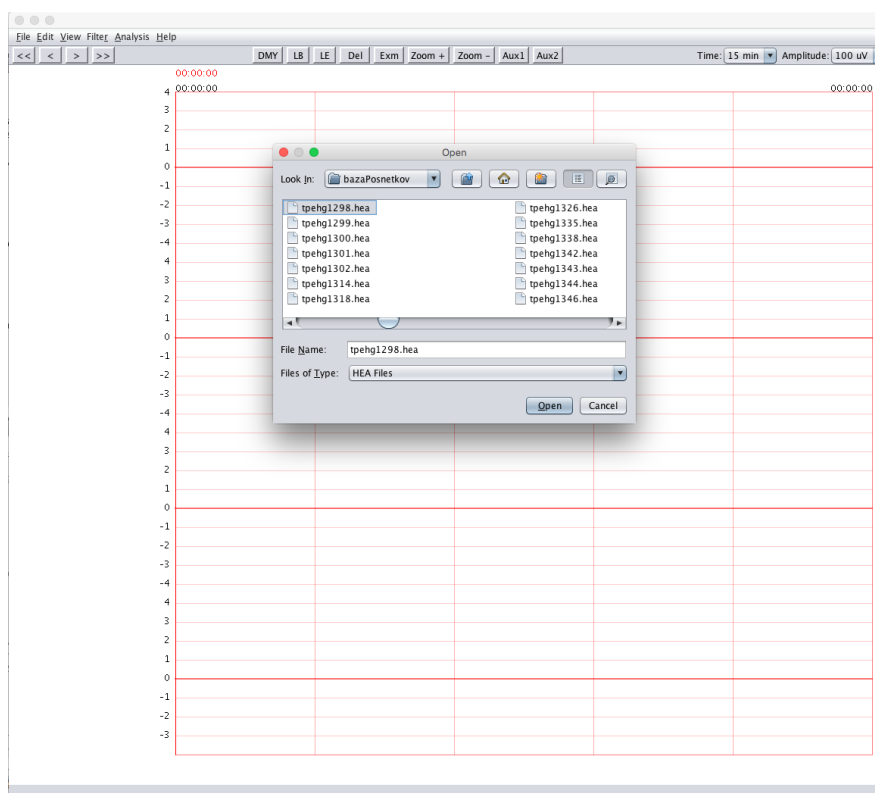
Z željo po hitrem načrtovanju smo se na začetku zanesli na papirnate prototipe. Tako smo na papir skicirali približno grafično podobo vmesnika, katero smo potem z različnimi popravki upoštevali pri dejanskem načrtovanju. Na sliki 4.1 je zadnja verzija prototipa na papirju pred dejanskim načrtovanjem.



Slika 4.1: Papirnati prototip

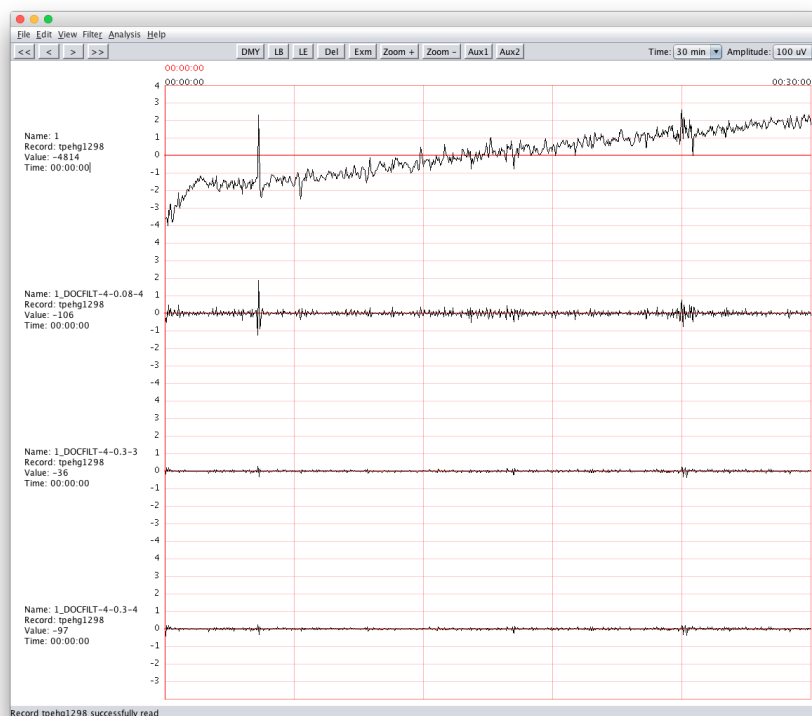
V nadaljevanju poglavja sledi opisna in slikovna predstavitev grafičnega vmesnika EHGLab skozi tipičen primer uporabe. Testne posnetke smo pridobili iz podatkovne baze TPEHG DB.

Ob zagonu vmesnika je za kakršno koli nadaljnje delo potrebno prebrati želen posnetek. Ob kliku na končno izbiro *Open* znotraj menuja *File* se nam odpre dialog na sliki 4.2 za izbiro želenega posnetka. Izbira *HEA Files* pod napisom *Files of Type*, nam prikaže vse datoteke s končnico *.hea*. To pomeni, da med naborom posnetkov dejansko izbiramo med njihovimi za-glavnimi datotekami. V našem primeru smo izbrali posnetek prezgodnjega poroda, ki se je zgodil v 36. tednu z imenom *tpehg1298*, kateri je bil zajet v 32. tednu nosečnosti.



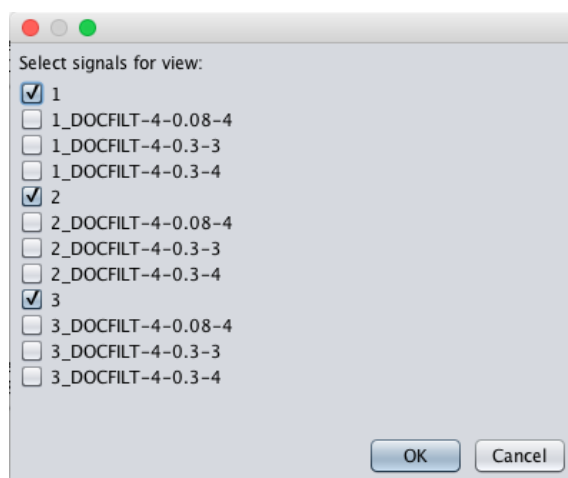
Slika 4.2: Dialog za odpiranje posnetka

Ob uspešnem branju se nam na risalno površino izrišejo prvi štirje signali (slika 4.3) v enakem zaporedju kot so zapisani v zaglavni datoteki.



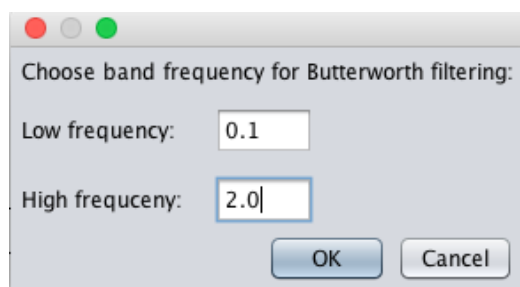
Slika 4.3: Prikaz signalov po uspešno prebranem posnetku

Denimo, da želimo pregledati vse tri kanale posnetka, to storimo preko končne izbire *Signals* znotraj menuja *View*. Prikaže se nam dialog z vsemi signali (slika 4.4), ki so znotraj našega posnetka. V našem primeru izberemo signale z imeni 1, 2, 3, kjer vsak signal predstavlja svoj kanal. Za pregled nad celotno dolžino posnetka spremenimo časovno skalo tako, da v orodni vrstici iz kombiniranega izvlečenega seznama *Time* izberemo vrednost 30min.

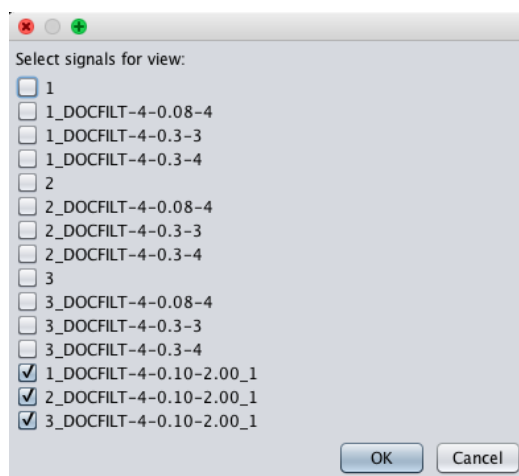


Slika 4.4: Dialog za izbiro izrisanih signalov

Ker vemo, da so pri posnetkih EHG lahko prisotni tudi šumi, jih bomo za nadaljnjo delo poizkušali odstraniti z digitalnimi Butterworthovimi filtri. Filtre bomo načrtali preko menuja *Filter*. Znotraj menuja *Filter* bomo izbrali končno izbiro *Custom Butterworth*, ki nam odpre dialog za vnos mejnih frekvenc (slika 4.5). V našem primeru smo za mejne frekvence izbrali vrednosti 0.1Hz in 2Hz . Po potrditvi vnosa mejnih frekvenc se opravi filtriranje. Filtrirani signali se dodajo k posnetku in so tako na voljo tudi za prikaz. Za izris na risalni površini jih izberemo preko končne izbire *Signals* znotraj menuja *View* (slika 4.6).

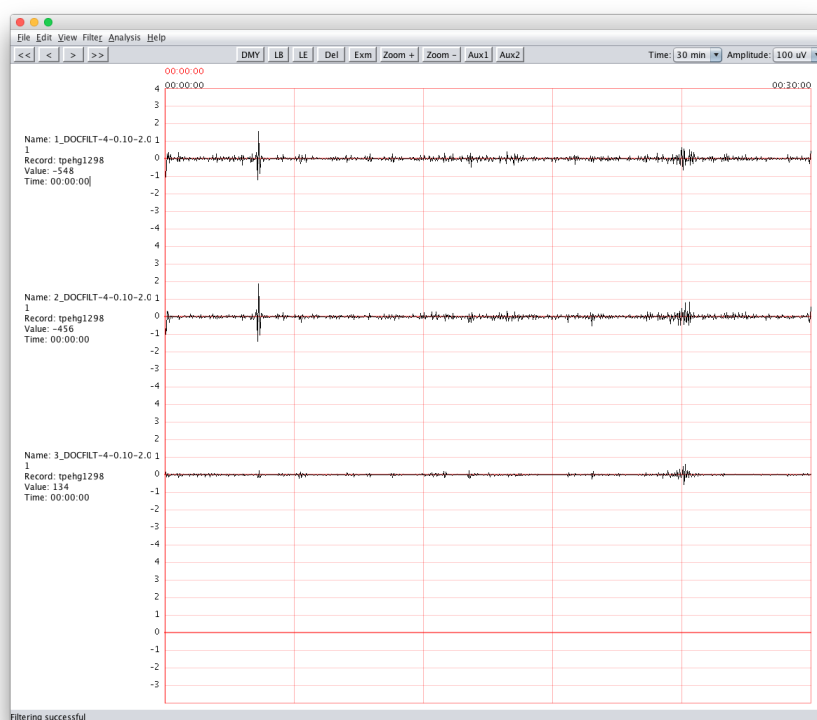


Slika 4.5: Dialog za izbiro mejnih frekvenc digitalnih Butterworthovih filtrov



Slika 4.6: Izbira filtriranih signalov znotraj uporabniškega vmesnika

Na sliki 4.7 torej vidimo signale, ki smo jih filtrirali znotraj uporabniškega vmesnika. Opazimo tudi, da je pri privzeti amplitudni skali $100\mu V$ vidljivost signalov slaba zato jo ustrezno popravimo. V našem primeru smo v orodni vrstici iz kombiniranega izvlečenega seznama *Amplitude* izbrali vrednost $50\mu V$. Denimo, da nas zanima dogodek, ki se je zgodil nekje pri koncu posnetka. Preko postavljenih oznak dogodek označimo. Za nadaljnjo delo z označenim delom, ga označimo z lokalnim intervalom. Z miškinim klikom na mesto, kjer želimo postaviti oznako za začetek lokalnega intervala se nam postavi marker trenutnega položaja. V orodni vrstici kliknemo na gumb z imenom *LB*. Postopek ponovimo tudi za oznako konca lokalnega intervala, le da izberemo gumb z imenom *LE*. Tako dobimo označen lokalni interval, kot je prikazano na sliki 4.8. Sedaj ko imamo označen lokalni interval lahko s tem delom posnetka počnemo vrsto stvari. Določen lokalni interval lahko z gumbom *Zoom +* v orodni vrstici izrišemo čez celotno risalno površino (slika 4.9). Označen lokalni interval lahko tudi izrežemo za morebitno nadaljnjo obdelavo. Izrez naredimo preko končne izbire *Snip* znotraj menija *Edit*. Ob izrezu se ustvari nov posnetek v formatu WFDB. V zaglavno datoteko se doda tudi komentar, da gre za izrezan posnetek in začetni čas izreza. V

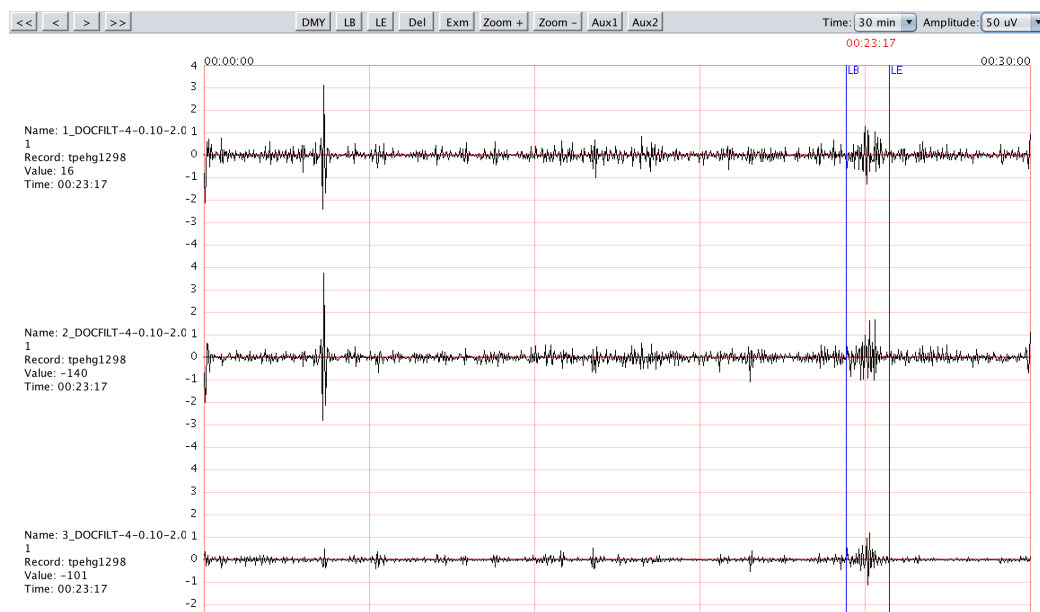


Slika 4.7: Prikaz filtriranih signalov znotraj uporabniškega vmesnika

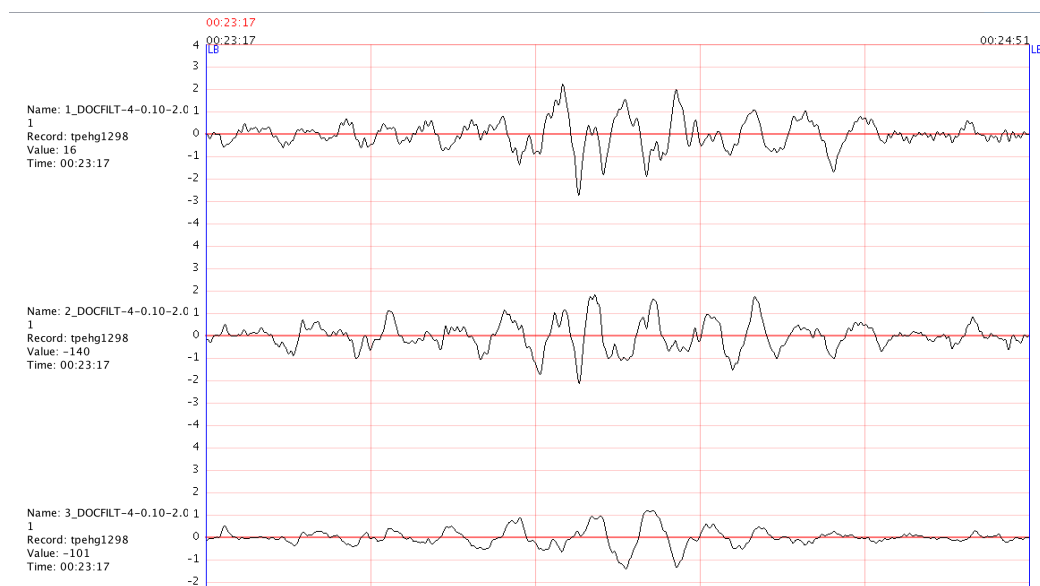
našem primeru se v zaglavno datoteko doda naslednji komentar:

```
#Produced by snip from record tpehg1298, beginning at 00:23:17
```

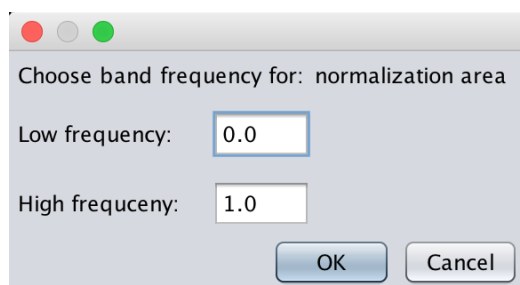
Ker smo lokalni interval označili zaradi nekaterih svojih lastnosti oziroma, ker bi lahko predpostavljali, da gre za popadek nas zanima kakšen je močnostni spekter in njegova analiza. V tem primeru bomo izračunali normaliziran močnostni spekter z algoritmom FFT. Znotraj menuja *Analysis* moramo najprej preko končne izbire *Normalization area* izbrati območje normalizacije (slika 4.10), v našem primeru smo normalizacijsko območje nastavili med 0Hz in 1Hz . Preko končne izbire *Analysis area* se odpre dialog za izbiro območja analize, katerega smo v našem primeru nastavili med 0Hz in 3Hz . Znotraj menuja *Analysis* bomo izbrali končno izbiro *Normalized Power spectrum* (slika 4.11).



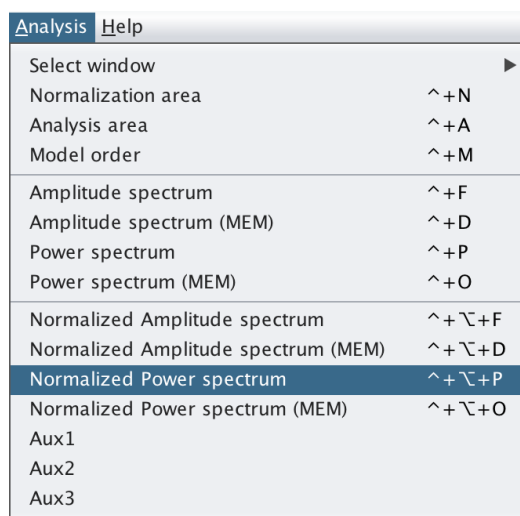
Slika 4.8: Označevanje lokalnega intervala



Slika 4.9: Prikaz povečanega dela signala znotraj lokalnega intervala

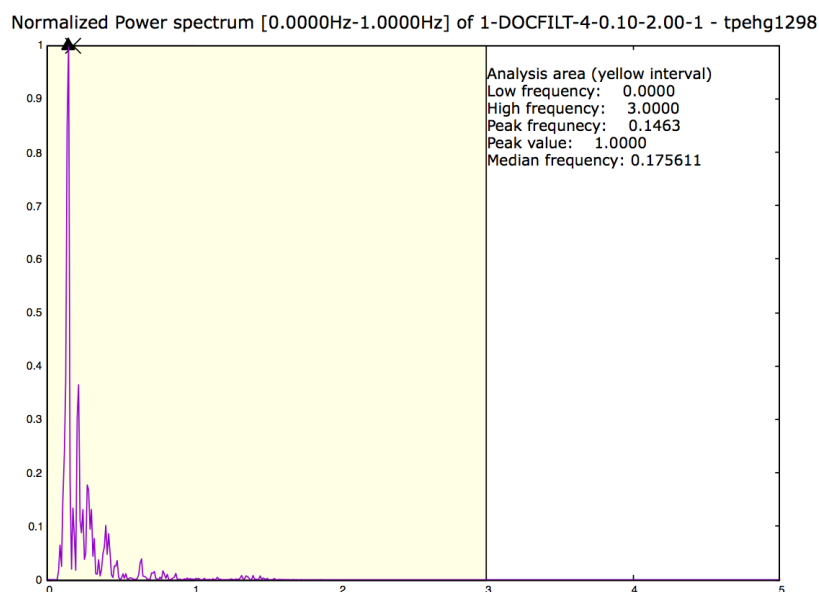


Slika 4.10: Dialog za vnos mejnih frekvenc območja normalizacije



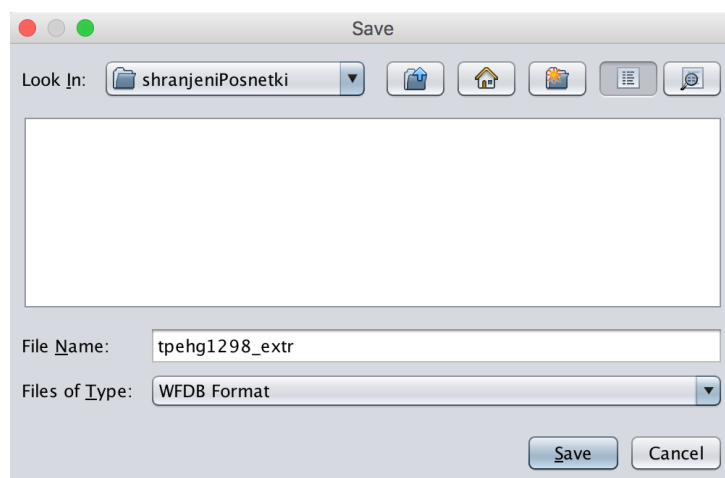
Slika 4.11: Menu *Analysis* in izbrana končna izbira za izračun normaliziranega močnostnega spektra

Ob izbiri končne izbire se izračunajo izbrani spektri za vsak signal, ki je prikazan na risalni površini. Vsi izračunani spektri se prikažejo v zunanjem oknu. V našem primeru so se tako prikazali trije izračunani spektri za vsak signal po en normaliziran močnostni spekter. Slika 4.12 prikazuje izračunan spekter za filtriran signal prvega kanala.



Slika 4.12: Normaliziran močnostni spekter filtriranega prvega kanala

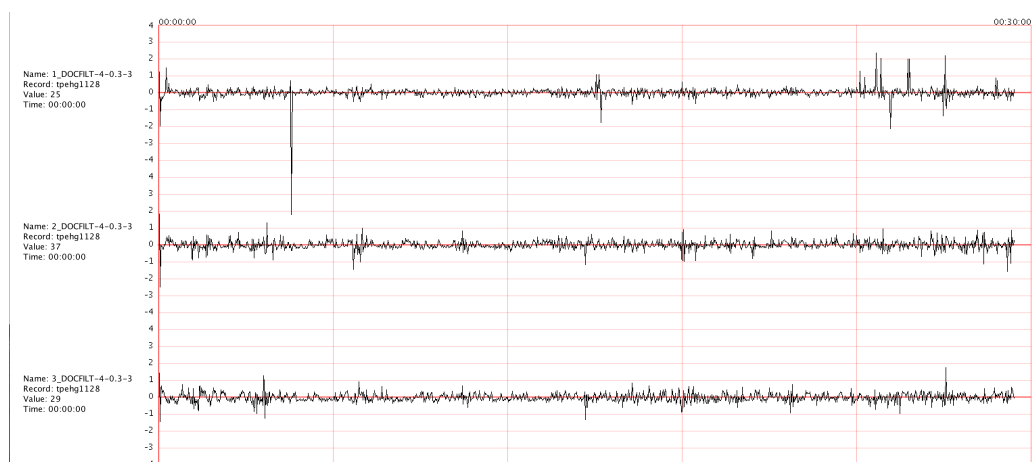
Ker smo znotraj uporabniškega vmesnika uporabili filtriranje, smo s tem ustvarili v našem primeru tri nove signale, ki so sedaj prisotni v posnetku. Lahko bi opravili tudi druga filtriranja in bi tako ustvarili še več novih signalov. V tem primeru lahko za nadaljnje delo celoten posnetek shranimo v nov posnetek v formatu WFDB ali v tekstovno datoteko. Pri shranjevanju v format WFDB se ustrezno ustvari tudi zaglavna datoteka, ki vsebuje informacije o novih signalih. Za shranjevanje lahko znotraj menija *File* uporabimo izbiro *Save as...*, ki nam ponudi končni izbiri za izbiro formata. V našem primeru smo izbrali shranjevanje v format WFDB. Ob kliku se prikaže dialog za shranjevanje (slika 4.13), kjer nam že predlaga morebitno novo ime. S klikom na gumb *Save* se posnetek shrani.



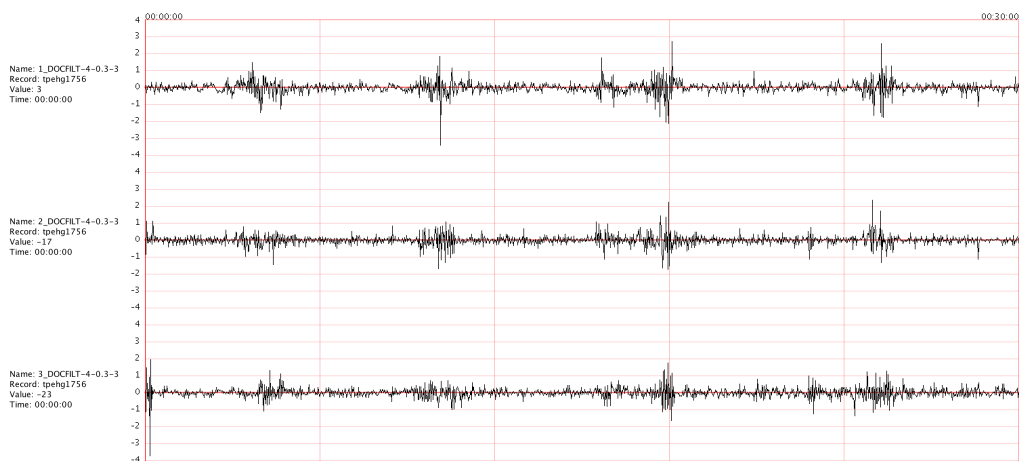
Slika 4.13: Dialog za shranjevanje posnetkov v formatu WFDB

V podpoglavju 1.2 smo v časovnem prostoru predstavili dva različna posnetka. Pri enem je šlo za terminski porod pri drugem pa za prezgodnji porod. Opazili smo nekaj razlik v signalih, ki so bili predstavljeni v časovnem prostoru. V nadaljevanju bomo poizkušali podobne razlike pogledati v našem uporabniškem vmesniku ter analizirati njihove močnostne spektre.

Za potrebe analize smo izbrali dva posnetka iz podatkovne baze TPEHG DB zajeta v 32. tednu in sicer *tpehg1128*, kjer je šlo za terminski porod in *tpehg1756*, kjer je šlo za prezgodnji porod. Posnetke smo najprej filtrirali z digitalnimi Butterworthovi filtri z dvosmerno shemo filtriranja mejnih frekvenc $0.3Hz$ in $3Hz$. Na risalno površino tako izrišemo tri signale, kjer vsak predstavlja svoj kanal. Pri sliki 4.14, ki predstavlja terminski porod lahko opazimo res nekoliko bolj kaotične signale. Pri sliki 4.15, ki predstavlja prezgodnji porod pa opazimo nekoliko bolj predvidljivo obliko oziroma lažje določimo periodo mišičnih aktivnosti. V podpoglavju 1.2 smo govorili, da se iz časovne predstavitve zdi, da je pri nosečnostih, kjer je šlo za terminski porod vpletenih nekoliko več fizioloških mehanizmov kot pri prezgodnjem porodu.



Slika 4.14: Vsi kanali posnetka tpehg1128 predstavljeni v časovnem prostoru



Slika 4.15: Vsi kanali posnetka tpehg1756 predstavljeni v časovnem prostoru

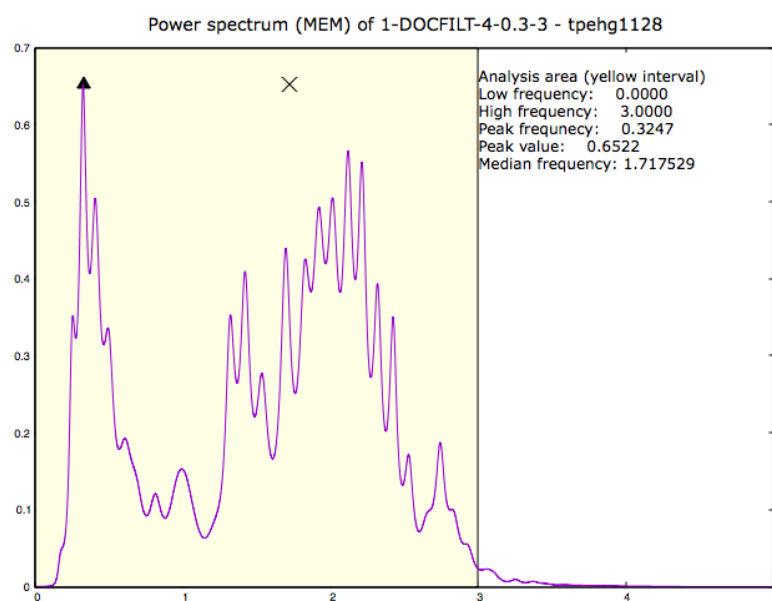
V uporabniškem vmesniku EHGLab lahko tako pogledamo tudi frekvenčno predstavitev posameznih posnetkov. Izračunali bomo močnostne spektre posameznih signalov in naredili primerjavo. Močnostne spektre bomo ocenili s pomočjo avtoregresivne metode. Za območje analize smo izbrali kar celotno območje prisotnih frekvenc. Ker opravljamo analizo filtriranih signalov smo za mejne frekvence izbrali $0.3Hz$ in $3Hz$.

Na slikah 4.16 in 4.17 sta predstavljena močnostna spektra prvega kanala terminskega in prezgodnjega poroda. Iz močnostnega spektra lahko razberemo, da pri terminskem porodu predstavlja manjšo moč v frekvenčnem pasu od $0Hz$ do $1Hz$. Bistveno razliko pa lahko opazimo v frekvenčnem pasu med $1Hz$ in $3Hz$. Razlika je opazna tudi v mediani frekvence. Pri terminskem porodu je mediana frekvence $1.72Hz$, pri prezgodnjem porodu pa $0.48Hz$.

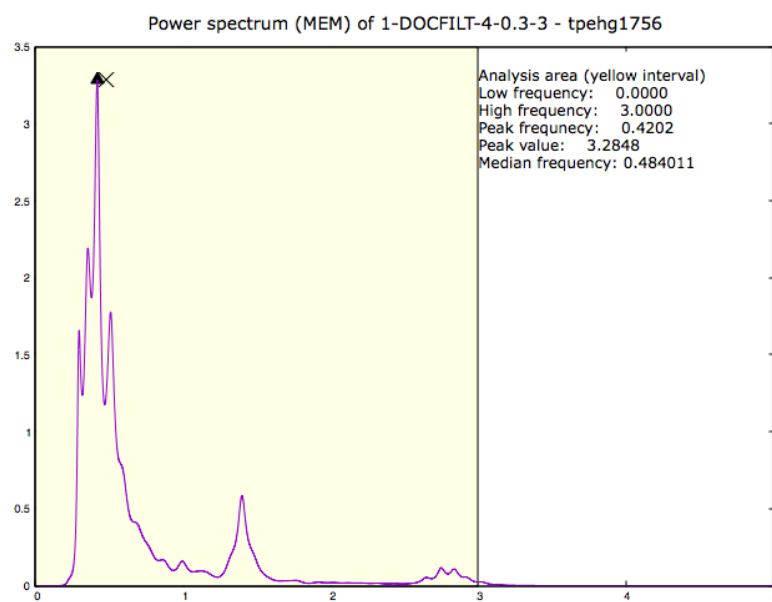
Podobno lahko opazujemo tudi na slikah 4.18 in 4.19, ki predstavljata močnostna spektra drugega kanala terminskega in prezgodnjega poroda. V tem primeru spet vidimo večjo moč pri prezgodnjem porodu v frekvenčnem pasu med $0Hz$ in $1Hz$. V drugem kanalu prezgodnjega poroda pa z razliko od prvega kanala niti ni opaziti posebne aktivnosti od $1Hz$ naprej, medtem, ko je pri terminskem porodu še vedno mogoče opaziti aktivnost od $1Hz$ naprej. Mediana frekvence drugega kanala pri terminskem porodu je $0.52Hz$ pri prezgodnjem porodu pa $0.42Hz$.

Enako velja za tretji kanal. Na slikah 4.20 in 4.21 sta predstavljena močnostna spektra tretjega kanala terminskega in prezgodnjega poroda. Ponovno je opaziti pri terminskem porodu nekoliko manjšo moč v frekvenčnem območju med $0Hz$ in $1Hz$ ter nekoliko močnejšo prisotnost frekvenc od $1Hz$ naprej. Mediana frekvence tretjega kanala pri terminskem porodu je $0.65Hz$ pri prezgodnjem porodu pa $0.46Hz$.

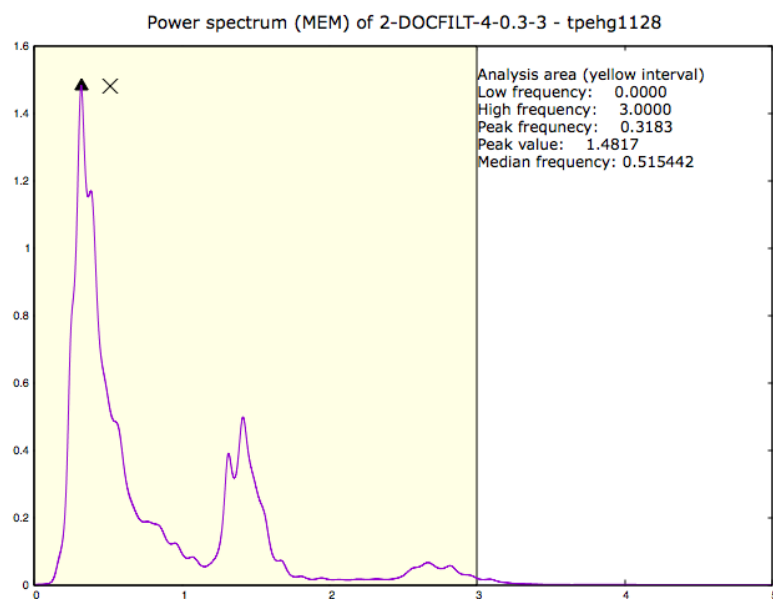
Na prikazanih močnostnih spektrih smo pokazali, kar je moč opaziti tudi v časovni predstavitvi signalov. Iz močnostnih spektrov lahko razberemo, da se frekvenca mišičnih aktivnosti maternice razlikuje med terminskim in prezgodnjim porodom.



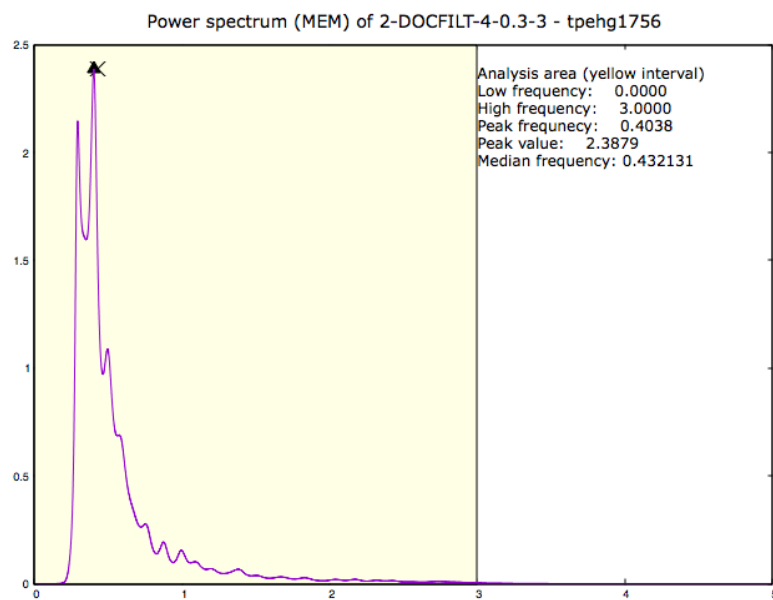
Slika 4.16: Močnostni spekter 1. kanala posnetka *tpehg1128* - terminski porod



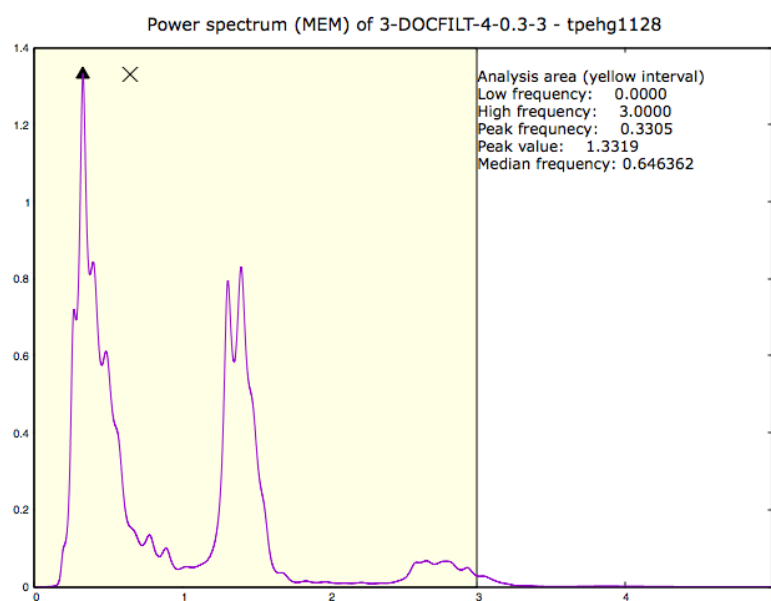
Slika 4.17: Močnostni spekter 1. kanala posnetka *tpehg1756* - prezgodnji porod



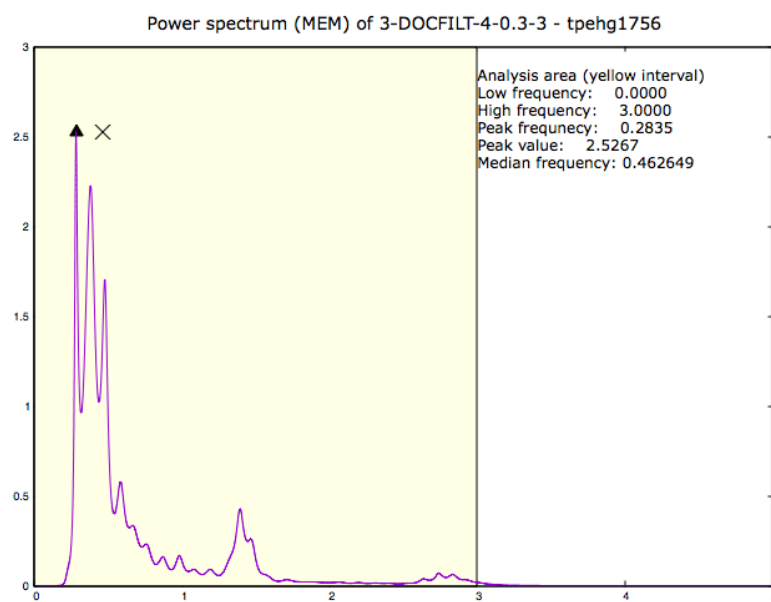
Slika 4.18: Močnostni spekter 2. kanala posnetka *tpehg1128* - terminski porod



Slika 4.19: Močnostni spekter 2. kanala posnetka *tpehg1756* - prezgodnji porod



Slika 4.20: Močnostni spekter 3. kanala posnetka *tpehg1128* - terminski porod



Slika 4.21: Močnostni spekter 3. kanala posnetka *tpehg1756* - prezgodnji porod

Poglavje 5

Razprava in zaključki

Razvili smo grafični uporabniški vmesnik za vizualizacijo, procesiranje in analizo posnetkov EHG, katerega smo imenovali EHGLab. Glavni namen je bil doseči višjo stopnjo uporabnosti pri delu s posnetki EHG. EHGLab omogoča branje in shranjevanje posnetkov v formatu WFDB in tako ni potrebe po različnih pretvorbah denimo v tekstovne datoteke. Za morebitne obdelave posnetkov v drugih programih prav tako omogočamo shranjevanje v tekstovno datoteko. Vsak signal, ki sestavlja prebran posnetek lahko v uporabniškem vmesniku grafično predstavimo v časovnem prostoru. Zaradi prisotnosti šumov v posnetkih EHG smo omogočili načrtovanje digitalnih Butterworthovih filtrov poljubnih mejnih frekvenc z dvosmerno shemo filtriranja. Pri načrtovanju filtrov smo uporabniku pustili proste roke pri izbiri reda filtrov. Prav tako smo nastavili hitre končne izbire s pred pripravljenimi mejnimi frekvencami. Za pred pripravljene mejne frekvence smo izbrali enake vrednosti, kot so jih izbrali razvijalci podatkovne baze TPEHG DB. Zagotovili smo tudi potrebo po osnovni manipulaciji. EHGLab tako omogoča postavljanje oznak, povečevanje označenih delov posnetkov, ter izrezovanje določenih posnetkov. EHGLab omogoča tudi pogled v frekvenčni prostor signalov. Za vsak signal lahko ocenimo in prikažemo amplitudne in močnostne spektre. Ocenjevanje spektrov lahko opravimo z uporabo algoritma FFT in avto regresivno metodo, ki prikaže nekoliko bolj zvezen spekter kot upo-

raba FFT. Nad vsakim spektrom lahko opravimo tudi normalizacijo znotraj poljubno določenih mejnih frekvenc. Za vrednotenje spektrov smo vključili tudi linearne tehnike analize spektrov. Vsak izračunan spekter lahko znotraj poljubnega območja analize ovrednotimo z iskanjem vrha spektra, frekvence vrha spektra in mediane frekvence.

Študije kažejo, da bi bila analiza posnetkov EHG lahko ena izmed tehnik za napovedovanje prezgodnjega poroda. Najboljšo mogočo analizo posnetkov bi lahko opravili, če posnetek ne bi vseboval kakršnih koli zunanjih vplivov oziroma šumov. Ker v praksi takšnih posnetkov ni mogoče zajeti je potrebno prisotne šume čim bolj izločiti. Zaradi izbire poljubnih mejnih frekvenc pri načrtovanju filtrov lahko znotraj vmesnika poizkušamo na hiter in enostaven način izločiti čim večji del prisotnih šumov. Zaradi možnosti sprotne analize pa bi lahko bilo orodje uporabno pri razvoju tehnik za napovedovanje prezgodnjega poroda.

Uporabniški vmesnik smo razvijali z mislijo dodajanja morebitnih novih funkcionalnosti. Tako smo na predvidenih mestih pustili prostor za nadaljnji razvoj. S tem smo zagotovili, da nadaljnji razvoj ne bi zahteval spreminjanja obstoječe arhitekture. Nadaljnje delo bi lahko obsegalo implementacijo dodatnih filtrov in vrsto različnih tehnik za analizo. Trenutno je v uporabniškem vmesniku omogočeno pregledovanje le enega prebranega posnetka. Za morebitno primerjavo različnih posnetkov bi lahko omogočili istočasen pregled več posnetkov hkrati. Z dodajanjem funkcionalnosti bi tako lahko dosegli še bolj učinkovito orodje za katerega ne izključujemo možnosti uporabe v klinični praksi.

Literatura

- [1] Spletni strežnik PhysioNet. <https://www.physionet.org/>.
- [2] Živa Novak, Vesna Fabjan Vodusek, and Lili Steblovnik. Zelo prezgodnji porod: presejalni testi za napovedovanje tveganja. *Medicinski Razgledi*, 47:367–378, 2008.
- [3] Ivan Verdenik. *Večplastni napovedni model prezgodnjega poroda: Multi-layer prediction model for preterm delivery : doktorsko delo*. Univerza v Ljubljani, Medicinska fakulteta, Ljubljana, 2002.
- [4] Charles M Steer and George J Hertsch. Electrical activity of the human uterus in labor. *American Journal of Obstetrics & Gynecology*, 59(1):25–40, 1950.
- [5] Hélène Leman, Catherine Marque, and Jean Gondry. Use of the electrohysterogram signal for characterization of contractions during pregnancy. *Biomedical Engineering, IEEE Transactions on*, 46(10):1222–1229, 1999.
- [6] Gašper Fele-Žorž, Gorazd Kavšek, Živa Novak-Antolič, and Franc Jager. A comparison of various linear and non-linear signal processing techniques to separate uterine EMG records of term and pre-term delivery groups. *Medical & biological engineering & computing*, 46(9):911–922, 2008.
- [7] NetBeans IDE spletna stran. <https://netbeans.org/>.

-
- [8] Programski paket WFDB. <https://www.physionet.org/physiotools/wfdb.shtml>.
- [9] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.
- [10] PhysioBank spletna stran. <https://www.physionet.org/physiobank/>.
- [11] George B Moody. WFDB Programmer’s Guide. *Harvard-MIT Division of Health Sciences and Technology*, 10, 2015.
- [12] George B Moody. WFDB Applications Guide. *Harvard-MIT Division of Health Sciences and Technology*, 10, 2015.
- [13] George B Moody. WAVE User’s Guide. *Harvard-MIT Division of Health Sciences and Technology*, 2015.
- [14] Gnuplot spletna stran. <http://www.gnuplot.info/>.
- [15] Java DSP collection spletna stran. <http://www.source-code.biz/dsp/java>.
- [16] Theo Mandel. *The elements of user interface design*, volume 20. Wiley New York, 1997.
- [17] Jakob Nielsen. Heuristic evaluation. *Usability inspection methods*, 17(1):25–62, 1994.
- [18] Franc Jager. Semi-automated interactive graphic tool to annotate 24-hour ambulatory ecg records. vabljeno predavanje. *Cambridge (Massachusetts): The Harvard - MIT, Division of Health Sciences and Technology*, 10, 2003.

-
- [19] Catherine Marque, Jacques MG Duchene, Sylvie Leclercq, Gilles S Pan-
czer, and Jean Chaumont. Uterine EHG processing for obstetrical moni-
torng. *Biomedical Engineering, IEEE Transactions on*, (12):1182–1187,
1986.
 - [20] Richard G Lyons. Understanding Digital Signal Processing. *A Prentice
Hall PTR Publication*, pages 49–51.
 - [21] Leif Sörnmo and Pablo Laguna. *Bioelectrical signal processing in cardiac
and neurological applications*. Academic Press, 2005.